arXiv:2602.07945v1 [math.NA] 8 Feb 2026

一种用于非线性偏微分方程的量子启发多层张量列车整体时空方法

N. R. RAP AKA ¶*, R. PEDDINTI†, E. TIUNOV†, N. J. FARAJ‡, A. N. ALKHOOORI‡, L. AOLITA†, Y. ADDAD *§, 和 M. K. RIAHI* ¶

摘要。 我们提出一种多层张量列车（TT）框架，用于在全局时空形式下求解非线性偏微分方程（PDE）。尽管时空张量列车求解器在压缩高维模拟方面展现出巨大潜力，但现有文献中与经典时间步进方法进行系统比较的研究很少，误差收敛分析有限，对张量列车舍入影响数值精度的定量评估也不多。同样，已有研究未能展示该方法在多种偏微分方程及参数范围下的性能。在实际应用中，对于强非线性、刚性或对流主导区域，整体牛顿迭代可能停滞或无法收敛；此时，较差的初始猜测以及严重病态的时空雅可比矩阵会妨碍稳健收敛。为克服这一局限，我们提出一种完全嵌入张量列车格式中的由粗到细多层策略。每一层都细化空间和时间分辨率，同时通过低秩延拓算子传递张量列车解，为后续牛顿求解提供稳健的初始值。残差、雅可比矩阵和传递算子均直接采用张量列车表示，并通过自适应秩密度矩阵重整化群算法求解。所提方法保留了张量列车表示的压缩能力和可扩展性，同时显著提高了收敛性和稳定性。针对若干非线性偏微分方程——包括 Fisher–KPP、粘性 Burgers 方程、正弦–戈登方程和 KdV——开展的数值实验涵盖了扩散、对流和色散动力学，结果表明，在单层时空牛顿迭代失败的情况下，多层张量列车方法仍能稳定收敛。在动态的对流主导非线性场景中，多层张量列车方法优于单层张量列车方法，能够以显著降低的计算成本实现高精度，尤其适用于要求高保真数值模拟的情形。

关键词。 量化张量列车、时空求解器、非线性偏微分方程

AMS主题分类。 68Q12, 68Q09 ,65F55 ,65M06 ,14N07 ,65M30

1. 引言。 非线性时变偏微分方程的数值解仍然是一项重大挑战，尤其是对于具有陡峭梯度、波动现象或刚性动力学的问题。本文提出并严格验证了一种新颖的计算框架，显著推进了此类问题的最先进水平。我们比较了三种不同的数值方法：作为基准的经典时间步进（CT）方法；代表整体式方案直接低秩推进的单层张量列车时空方法（SL-TT）；以及本文的主要贡献——多层张量列车时空方法（ML-TT）。

我们的方法以张量列车（TT）分解为数学基础，它为缓解高维问题中的维数灾难提供了强有力的框架。张量列车格式作为规范分解和 Tucker 分解的一种计算高效的替代方案而被提出，它将一个 d维张量表示为由低秩核构成的链 [20]。

* 数学系、计算与数学科学学院，哈利法大学科技学院，阿联酋阿布扎比。电子邮件：narsimha.rapaka@ku.ac.ae

† 技术创新研究所，阿布扎比，阿联酋 电子邮箱：raghavendra.peddinti@tii.ae, egor.tiunov@tii.ae, leandro.aolita@tii.ae。

‡ 阿联酋核能公司，马斯达尔城，阿布扎比，阿联酋。电子邮箱：noor.faraj@enec.ae§ 机械与核工程系，哈利法科技大学，阿布扎比，阿联酋 电子邮箱：yacine.addad@ku.ac.ae

¶ 阿联酋核技术中心，哈利法科技大学，阿布扎比，阿联酋 通信请寄至：mohamed.riahi@ku.ac.ae

后续发展包括用于黑箱张量逼近的 TT-cross 近似[18]，以及量化 TT（QTT）格式；后者通过将长向量折叠为高阶张量，实现了对数级压缩 [14]。这些进展使高维偏微分方程的高效求解成为可能 [7, 15]，并已与密度矩阵重整化群等迭代求解器 [26] 和 AMEN [8] 相结合，用于求解张量列车格式下的线性系统。

整体时空方法将演化问题表述为整个时空域上的单个耦合系统，为顺序时间步进提供了一种替代方案。通过同时求解所有时间层级，该方法避免了逐步积分所伴随的经典局部截断误差累积，并支持在整个时间范围内对非线性项进行统一的隐式处理。结合张量列车（TT）表示后，时空表述能够利用空间和时间维度上的可分离性，大幅压缩时空自由度并降低问题的有效维数。先前的研究已展示了这类方法的潜力，包括针对福克–普朗克方程的时空 TT 方法 [7], 、用于微分黎卡提方程的非线性方法 [6], ，以及通过谱方法/配置法和张量列车方法求解对流扩散系统和反应扩散系统的时空方法 [3, 2, 4]。此外，研究者还开发了在 TT 流形上采用逐步截断的隐式时间推进方法，以控制积分过程中的秩增长[25]。

然而，现有的单层 TT 时空方法往往存在关键局限。对于刚性非线性问题，其性能和稳健性可能对初始猜测高度敏感。此外，针对多种非线性偏微分方程，与经典时间步进方法的直接比较较少；将离散误差与 TT 舍入效应分离开来的系统误差研究也有限；而整体牛顿迭代在刚性或对流主导区域可能缺乏稳健性。这些方法在复杂非线性双曲区域中的表现尤其是一个显著的研究空白。例如，虽然粘性 Burgers 方程已在其抛物型区域得到研究，但尚未有人对其双曲型区域中的整体 TT 求解器进行全面的定量评估——在该区域，非线性对流占主导并导致激波形成。在此类区域，采用分块时间域分解以及时间并行稳定双共轭梯度算法 [24]，可能有助于求解双曲型问题，缓解高键维数引发的适定性问题，并通过并行计算进一步加速求解。此外，近期的量子启发式进展提出了一种时空求解器，可实现运行时间的对数级扩展，并通过 MPS-DMD [22] 等算法实现准确的数据驱动预测。这些方法凸显了全局时空表示能够以最少的自由度捕捉复杂时空相关性的有效性。

我们的工作直接针对这些不足。尽管单层 TT 方法（SL-TT）本身就是一项贡献，但我们的主要进展是多层 TT（ML-TT）框架。该方法利用由粗到细的时空网格层级，系统地生成高质量的初始猜测，从而显著提升牛顿求解器的鲁棒性和收敛速度。

本研究超越了概念验证，在广泛的非线性动力学问题中展示了经过验证的解：抛物型行为（例如，扩散主导区域中的 Fisher-KPP [10]和粘性 Burgers 方程）、双曲型行为（对流主导的激波形成区域中的粘性 Burgers 方程以及正弦-戈登方程 [5]），以及色散波传播（例如，科尔特韦格-德弗里斯方程 [29]）。针对每种情况，我们都进行了详细的网格细化研究，并分析了低秩截断的影响。ML-TT 方法始终优于经典方法和单层方法，在稳定收敛方面表现尤为突出，尤其适用于

具有挑战性的问题。该方法成功应用于广泛的一类非线性问题，是本文的一项关键成果，证明了它能够处理强非线性和陡峭梯度；而现有整体低秩求解器尚未经过充分测试，无法确定其是否具备这种能力。

乍看之下，所提出的多层策略似乎与经典时空多重网格方法有关 [13, 23]。然而，当前方法在目的和构造上都截然不同。经典多重网格方法作用于线性化算子，并依靠残差限制、粗网格校正和循环策略来加速线性求解器的收敛。相比之下，我们的多层框架仅应用于非线性层面，其唯一用途是在逐步细化的时空网格上，为整体牛顿迭代生成高质量的初始猜测。此方法不进行残差传递、不将残差限制到粗网格，也不进行多层循环；每一层都独立求解至收敛，随后将解延拓至下一层更细的网格，作为牛顿法的初始猜测。这一区别在张量列车场景中至关重要，因为主要的失效模式是非线性停滞或发散，而非线性收敛缓慢。

整体形式允许灵活选择时间积分器。我们的张量列车形式既包括稳健性较强的一阶隐式欧拉格式，也包括在需要更高精度以重新求解波动力学时采用的二阶时间格式，例如克兰克-尼科尔森格式和纽马克方法。由所有这些格式产生的全局系统都能无缝纳入张量列车框架。

本文组织如下：第2 节介绍张量列车格式的必要背景。第3 节详细说明整体式时空离散化、牛顿求解器以及新颖的多层算法。第4 节展示了我们广泛开展的数值实验，对比了三种方法在不同 PDE 类别和情形下的表现。最后，第5 节给出总结性评述。

2. 张量列车（TT）方法。张量列车（TT）格式是一种用于高维数组的可分离表示，它通过将一个 d维张量表示为由小型三维“核”构成的链，缓解维数灾难。设 $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ 的元素为 $\mathcal{A}(i_1, \ldots, i_d)$。张量列车表示写作

$$\mathcal{A}(i_1, \ldots, i_d) = \sum_{\alpha_1=1}^{r_1} \cdots \sum_{\alpha_{d-1}=1}^{r_{d-1}} G^{(1)}(1, i_1, \alpha_1)\, G^{(2)}(\alpha_1, i_2, \alpha_2) \cdots G^{(d)}(\alpha_{d-1}, i_d, 1), \tag{2.1}$$

其中，张量列车核为 $G^{(k)} \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$，且 $r_0 = r_d = 1$。整数 r_k 称为张量列车秩（也称键维数）；较小的秩意味着较强的可分离性，并可实现显著压缩。沿秩索引收缩相邻核，即秩-核乘积，记作 $\bowtie$，是张量列车运算的基本构件。对于两个连续的核，

$$\big(G^{(k)} \bowtie G^{(k+1)}\big)(\alpha_{k-1}, i_k, i_{k+1}, \alpha_{k+1}) = \sum_{\alpha_k=1}^{r_k} G^{(k)}(\alpha_{k-1}, i_k, \alpha_k)\, G^{(k+1)}(\alpha_k, i_{k+1}, \alpha_{k+1}).$$

因此 $\bowtie$ 将大小为 r_k的键收缩。键维数 $\chi = \max_k r_k$ 同时控制表达能力（允许的相关性）和计算成本。对于秩-核乘积 $\bowtie$，

$$\mathcal{A}(i_1, \ldots, i_d) = G^{(1)}(:, i_1, :) \;\bowtie\; G^{(2)}(:, i_2, :) \;\bowtie\; \cdots \;\bowtie\; G^{(d)}(:, i_d, :).$$

矩阵 $A \in \mathbb{R}^{N\times M}$ 在 $N = \prod n_k$ 和 $M = \prod m_k$ 的情况下，使用张量列矩阵（又称矩阵乘积算子）核 $M^{(k)} \in \mathbb{R}^{r_{k-1}\times n_k\times m_k\times r_k}$：

$$A\big((i_1,\dots,i_d),(j_1,\dots,j_d)\big) = \sum_{\alpha_1,\dots,\alpha_{d-1}} \prod_{k=1}^{d} M^{(k)}(\alpha_{k-1}, i_k, j_k, \alpha_k).$$

张量列矩阵乘以张量列向量后仍保持为张量列车格式，其舍入前的秩通过逐分量相乘得到：$r_k^{Ax} \le r_k^A\, r_k^x$。

2.1. 张量列车格式中的基本代数。 张量列车（TT）格式中的代数运算可以通过张量列链的逐核组合来表示。设

$$\mathcal{X} = G^{(1)} \bowtie \cdots \bowtie G^{(d)}, \qquad \mathcal{Y} = H^{(1)} \bowtie \cdots \bowtie H^{(p)}, \qquad A = M^{(1)} \bowtie \cdots \bowtie M^{(d)}$$

为张量列车格式中的张量，其张量列车秩分别为 $r_k^{\mathcal{X}}$、$r_k^{\mathcal{Y}}$ 和 r_k^A，其中 $r_0^{\mathcal{X}} = r_d^{\mathcal{X}} = r_0^{\mathcal{Y}} = r_p^{\mathcal{Y}} = r_0^A = r_d^A = 1$。此处，$\mathcal{X}$ 和 $\mathcal{Y}$ 是每个核只有一个物理指标的张量列车向量，而 A 是一个张量列矩阵（也称为矩阵乘积算子），其核 $M^{(k)}$ 带有两个物理指标，分别对应输入和输出模态。对于加法、逐元素乘法、矩阵-向量乘积或矩阵-矩阵乘积，张量的物理维度必须兼容，即模态大小相同且 $d = p$。对于克罗内克积，物理维度可以不同，并且 d 不必等于 p。

运算	核心构造	秩的行为
加法（$\mathcal{X}+\mathcal{Y}$）	块对角拼接核	of $r_k = r_k^{\mathcal{X}} + r_k^{\mathcal{Y}}$；TT 舍入必需
逐元素乘积（$\mathcal{X}\odot\mathcal{Y}$）	（阿达马积）按秩克罗内克耦合具有共享物理指标的核	$r_k \le r_k^{\mathcal{X}} r_k^{\mathcal{Y}}$；TT 舍入必需
矩阵-向量（$A\mathcal{X}$）	/物理指标上的局部缩并	$r_k \le r_k^A r_k^{\mathcal{X}}$；TT 舍入必需
矩阵-矩阵乘积	物理指标	
克罗内克积（$\mathcal{X}\otimes\mathcal{Y}$）	通过 a 连接 TT 核秩一键	无秩增长；无需进行轮转

表 1：张量列车（TT）格式中的基本代数运算。

表1总结了主要的 TT 代数运算及其秩行为。加法会导致秩相加增长，而逐元素乘积和基于矩阵的乘积则通过克罗内克积类型的运算或缩并运算引入乘法式秩增长。在所有这类情况下，都需要进行 TT 舍入以控制秩。相比之下，克罗内克积可通过将各操作数的 TT 核借助平凡的秩一键串接起来，得到精确的张量列车表示。表 1 中列出的运算定义如下。

$$\begin{aligned}
\mathcal{X}+\mathcal{Y} &= \begin{bmatrix} G^{(1)} & H^{(1)} \end{bmatrix} \bowtie \begin{bmatrix} G^{(2)} & 0 \\ 0 & H^{(2)} \end{bmatrix} \bowtie \cdots \bowtie \begin{bmatrix} G^{(d)} \\ H^{(d)} \end{bmatrix}, \\
\mathcal{X}\odot\mathcal{Y} &= \big(G^{(1)} \otimes_{\mathrm{r}} H^{(1)}\big) \bowtie \cdots \bowtie \big(G^{(d)} \otimes_{\mathrm{r}} H^{(d)}\big), \\
A\mathcal{X} &= \big(M^{(1)} \star G^{(1)}\big) \bowtie \cdots \bowtie \big(M^{(d)} \star G^{(d)}\big), \\
\mathcal{X}\otimes\mathcal{Y} &= G^{(1)} \bowtie \cdots \bowtie G^{(d)} \bowtie H^{(1)} \bowtie \cdots \bowtie H^{(p)}.
\end{aligned}$$

这里， $\otimes_{\mathrm{r}}$ 表示对 TT 秩指标取克罗内克积，并针对每个物理指标独立应用；而逐核收缩积 $\star$ 表示对 TT 核之间共享的物理指标进行局部缩并，同时将相关的 TT 秩指标相乘组合；这两种运算均在下文明确给出定义，同样的构造也适用于 TT-matrix $\times$ TT-matrix 乘法。

$$\big(G^{(k)} \otimes_{\mathrm{r}} H^{(k)}\big)(\alpha_{k-1}\beta_{k-1},\, i_k,\, \alpha_k\beta_k) = G^{(k)}(\alpha_{k-1}, i_k, \alpha_k)\, H^{(k)}(\beta_{k-1}, i_k, \beta_k).$$

$$\big(M^{(k)} \star G^{(k)}\big)(\alpha_{k-1}\beta_{k-1},\, i_k,\, \alpha_k\beta_k) = \sum_{j_k=1}^{n_k} M^{(k)}(\alpha_{k-1}, i_k, j_k, \alpha_k)\, G^{(k)}(\beta_{k-1}, j_k, \beta_k).$$

2.2. 舍入与内存压缩。TT 张量的存储量为

$$\mathrm{nnz}(\mathrm{cores}) = \sum_{k=1}^{d} r_{k-1}\, n_k\, r_k \;\approx\; \mathcal{O}(d\, n\, r^2) \quad (\text{if } n_k \equiv n,\; r_k \equiv r).$$

当秩保持适中时，这将指数级的 $\mathcal{O}(n^d)$ 开销降至近乎线性的 d 扩展。完成代数运算后，*TT* 舍入（一系列 QR+SVD 截断）会降低秩，以满足容差 ε，其计算量为 $\mathcal{O}(d\, n\, r^3)$。

2.3. 量化张量列车（QTT）。QTT 会先将较大的物理指标折叠为多个较小的（通常为二进制）模式，再应用 TT，从而进一步压缩长向量和矩阵。对于长度为 N 的向量，选取基数 b（通常为 $b = 2$），并写作 $N = b^q$；指标 $i \in \{0, \ldots, N-1\}$ 具有 $i =$ 个以 b 为基数的数字 ι_1，…，ι_q，其中 $\iota_k \in \{0, \ldots, b-1\}$。折叠映射 $\mathcal{F}_b\colon \mathbb{R}^N \to \mathbb{R}^{b\times\cdots\times b}$（$q$ 个模式）将 x 重塑为 $\tilde{x}$（ι_1，…，ι_q）$= x$（i）。对 $\tilde{x}$ 应用 TT 后，得到一个核大小为 $r_{k-1} \times b \times r_k$、存储量为

$$\mathcal{O}\big(q\, b\, r^2\big) = \mathcal{O}\big((\log_b N)\, b\, r^2\big),$$

当秩保持有界时，其相对于原始长度 N 呈对数级增长。对于具有 $N = b^q$ 和 $M = b^p$ 的矩阵 $A \in \mathbb{R}^{N\times M}$，将行和列索引都折叠，得到具有 $q + p$个小模态（每个模态大小为 b）的 QTT 矩阵。许多结构化向量和算子（例如指数函数、正弦函数、Toeplitz／拉普拉斯型算子）都具有较小的 QTT 秩，从而能够大幅压缩并实现快速运算。所有基本运算（加法、阿达马积、矩阵向量乘法）都与张量列车中的运算相同，临时的秩增长则通过 QTT 舍入加以控制。

张量列车的存储量为 $\mathcal{O}(d\, n\, r^2)$，而当基数 b 较小时，QTT 的存储量为 $\mathcal{O}((\log_b N)\; r^2)$；这使得低秩利用更加稳健，并实现内存压缩，从而让张量列车／QTT 成为处理高维偏微分方程及相关问题的实用工具。

3. 多层 TT 时空方法。我们考虑标量场 $u = u(\mathbf{x}\; t)$ 的一般非线性初边值问题，其中 $\mathbf{x} \in \Omega \subset \mathbb{R}^d$，$t \in (0, T]$:

$$\frac{\partial^m u}{\partial t^m} + \mathcal{L}u + \mathcal{N}(u) = s(\mathbf{x}, t), \qquad \mathbf{x} \in \Omega,\; t \in (0, T], \tag{3.1}$$

$$\mathcal{B}(u, \nabla u, \ldots) = g(\mathbf{x}, t), \qquad \mathbf{x} \in \partial\Omega,\; t \in (0, T], \tag{3.2}$$

$$u^j(\mathbf{x}, 0) = u_0^j(\mathbf{x}), \qquad \mathbf{x} \in \Omega, \quad j = 0, 1, \ldots, m-1, \tag{3.3}$$

其中，$\mathbf{x} = (x_1, x_2, \ldots, x_d)^\top \in \mathbb{R}^d$ 是空间坐标，$\mathcal{L}$ 是线性空间微分算子，$\mathcal{N}(u)$ 是作用于 u 的非线性空间微分算子，s 是源项，$\mathcal{B}$ 是边界

算子（狄利克雷、诺伊曼、罗宾或周期），且 $u_0(\mathbf{x})$ 是初始条件。在本文中，我们考虑 $m = 1,2$ 的情形。

为简化描述，我们考虑一维空间$\mathbf{x} = x \in [x_a, x_b]$，将当前方法推广至多个空间维度也十分直接。将时空域离散为 $N_x \equiv 2^{q_x}$ 个和 $N_t = 2^{q_t}$ 个等间距单元或区间，其间距分别为 $\Delta x = (x_b - x_a)/N_x$ 和 $\Delta t = T/N_t$，并令 $q = q_x + q_t$。令 $x_i = x_a + i\Delta x/2, i = 1, 2, \ldots, N_x$ 为每个单元的中心，$t_n = n\Delta t, n = 1,2, \ldots, N_t$ 为每个时间区间的端点。定义 $u_i^n := u(x_i, t_n)$，并将时刻 t_n 的空间解向量记为 $U^n = [u_1^n, u_2^n, \ldots, u_{N_x}^n]^\top \in \mathbb{R}^{N_x}$，将时空解向量记为 $\mathcal{U} = [U^{1\top}, U^{2\top}, \ldots, U^{N_t\top}]^\top \in \mathbb{R}^{N_t \cdot N_x}$。同理，令 $s_i^n = s(x_i, t_n)$，并将时刻的空间源向量记为 $S^n = [s_1^n, s_2^n, \ldots, s_{N_x}^n]^\top \in \mathbb{R}^{N_x}$，将时空源向量记为 $\mathcal{S} = [S^{1\top}, S^{2\top}, \ldots, S^{N_t\top}]^\top \in \mathbb{R}^{N_t \cdot N_x}$。令 $\mathcal{L}_\Delta$ 和 $\mathcal{N}_\Delta$ 为离散化的线性算子和非线性算子，并定义

$$F(U, S) := \mathcal{L}_\Delta U + \mathcal{N}_\Delta(U) - S.$$

为便于施加边界条件，我们引入位于计算域外部的虚单元（左侧索引为 0，右侧索引为 $N_x + 1$），其宽度与内部单元相同。虚单元的值 u_0 和 u_{N+1} 可由内部单元的值 u_1 和u_N以及边界条件（以下标 b表示）表示。

$$u_0 = a_1 u_1 + b_1, \qquad u_{N+1} = a_2 u_N + b_2, \tag{3.4}$$

$$a_1, a_2 = \begin{cases} -1 & \text{Dirichlet BC} \\ +1 & \text{Neumann BC} \\ 0 & \text{no BC} \end{cases}, \quad b_1, b_2 = \begin{cases} 2u_b & \text{Dirichlet BC} \\ \frac{\partial u}{\partial n}\big|_b \Delta n & \text{Neumann BC} \\ 0 & \text{no BC} \end{cases}, \tag{3.5}$$

其中，a_1 和 a_2 取决于左右边界处的边界条件类型；b_1 和 b_2取决于左右边界处的非齐次边界值。这里，$\frac{\partial u}{\partial n} = \nabla u \cdot \hat{n}$、$\Delta n$和 $\hat{n}$ 分别表示网格间距和沿外法线方向的单位向量。

3.1. 整体时空离散设置。 时空离散使用以下向量和矩阵：$\mathbf{e}_1 = [1, 0, \ldots, 0]^\top \in \mathbb{R}^{N_t}$、$\mathbf{e}_2 = [0, 1, 0, \ldots, 0]^\top \in \mathbb{R}^{N_t}$，$I_x \in \mathbb{R}^{N_x \times N_x}$，$I_t \in \mathbb{R}^{N_t \times N_t}$ 为单位矩阵，$D_t \in \mathbb{R}^{N_t \times N_t}$ 是一个对角线元素为 1、次对角线元素为 -1 的下双对角托普利兹矩阵，$D_{tt} \in \mathbb{R}^{N_t \times N_t}$ 是一个对角线元素为 1、第一条次对角线元素为 -2 、第二条次对角线元素为 1 的下三对角托普利兹矩阵，$J_t \in \mathbb{R}^{N_t \times N_t}$是一个对角线元素为 $\mathbf{e}_2$、次对角线元素为 $\mathbf{e}_2$ 的下双对角托普利兹矩阵，而 $\mathcal{K}_t \in \mathbb{R}^{N_t \times N_t}$是一个对角线元素为 $\mathbf{e}_4$、第一条次对角线元素为 $\mathbf{e}_2$、第二条次对角线元素为 $\mathbf{e}_4$ 的下三对角托普利兹矩阵。所有算子的矩阵表示和张量列车表示见附录A。

以下所有离散化均得到关于时空解向量 $\mathcal{U}$ 的非线性系统，并用牛顿法求解：

$$f(\mathcal{U}) = A(\mathcal{U}) + B\mathcal{U} - \mathcal{C} = 0 \tag{3.6}$$

其中，$A(\mathcal{U})$ 是非线性的，B 是线性的，而 $\mathcal{C}$ 汇总了初始条件、源项以及非齐次边界条件带来的所有贡献；这些分量针对每种情况分别定义。

对于 $m = 2$，$\frac{\partial^2 u}{\partial t^2}$ 的离散近似需要 $U^{(-1)}$，即$t = -\Delta t$处的空间向量；该向量通过围绕 $t = 0$ 进行截断泰勒展开来数值近似，以保持 U_{tt}

与时间离散的精度保持一致。对于 $\frac{\partial^2 u}{\partial t^2}$ 的二阶离散化，$U^{(-1)}$ 近似为：

$$U^{(-1)} = U^0 - \Delta t\, U_t^0 + \frac{\Delta t^2}{2} U_{tt}^0 - \frac{\Delta t^3}{6} U_{ttt}^0 + \mathcal{O}(\Delta t^4), \tag{3.7}$$

其中，时间导数 U_{tt}^0和 U_{ttt}^0通过对方程 (3.1求导并代入初始条件 U^0和 U_t^0来评估。

3.1.1. 一阶时间离散化。 对于 $m = 1$，对方程 (3.1进行隐式欧拉离散，得到：

$$\frac{U^n - U^{n-1}}{\Delta t} + F(U^n, S^n) = 0, \quad n = 1, 2, \ldots, N_t.$$

完整的时空解向量 $\mathcal{U}$ 由以下系统求得：

$$\frac{1}{\Delta t}\left[(D_t \otimes I_x)\mathcal{U} - \mathbf{e}_1 \otimes U^0\right] + (I_t \otimes \mathcal{L}_\Delta)\mathcal{U} + (I_t \otimes \mathcal{N}_\Delta)(\mathcal{U}) = \mathcal{S}.$$

将上述方程改写为方程 (3.6) 的形式：

$$\underbrace{\Delta t(I_t \otimes \mathcal{N}_\Delta)}_{A}(\mathcal{U}) + \underbrace{(D_t \otimes I_x + \Delta t I_t \otimes \mathcal{L}_\Delta)}_{B}\mathcal{U} - \underbrace{(\mathbf{e}_1 \otimes U^0 + \Delta t \mathcal{S})}_{\mathcal{C}} = 0. \tag{3.8}$$

对于 $m = 2$，方程 (3.6) 包含两个初始条件 U^0和 U_t^0，并包含二阶导数算子 D_{tt}：

$$\underbrace{\Delta t^2(I_t \otimes \mathcal{N}_\Delta)}_{A}(\mathcal{U}) + \underbrace{(D_{tt} \otimes I_x + \Delta t^2 I_t \otimes \mathcal{L}_\Delta)}_{B}\mathcal{U} - \underbrace{(\mathbf{e}_1 \otimes (2U^0 - U^{(-1)}) - \mathbf{e}_2 \otimes U^0 + \Delta t^2 \mathcal{S})}_{\mathcal{C}} = 0, \tag{3.9}$$

其中，$U_{(-1)} = U^0 - \Delta t U_t^0 + \frac{\Delta t^2}{2} U_{tt}^0$，其中 U_{tt}^0由方程 (3.1) 根据初始数据 U^0 计算得到。

3.1.2. 二阶时间离散化。 对于 $m = 1$，方程 (3.1) 采用克兰克-尼科尔森格式进行离散化：

$$\frac{U^n - U^{n-1}}{\Delta t} + \frac{1}{2}\left[F(U^n, S^n) + F(U^{n-1}, S^{n-1})\right] = 0, \quad n = 1, 2, \ldots, N_t.$$

完整的时空解向量 $\mathcal{U}$ 满足非线性系统：

$$\frac{1}{\Delta t}\left[(D_t \otimes I_x)\mathcal{U} - \mathbf{e}_1 \otimes U^0\right] + (J_t \otimes \mathcal{L}_\Delta)\mathcal{U} + (J_t \otimes \mathcal{N}_\Delta)(\mathcal{U}) - (J_t \otimes I_x)\mathcal{S} + \frac{1}{2}(\mathbf{e}_1 \otimes F(U^0, S^0)) = 0.$$

将上面的方程改写为方程（3.6）：

$$\underbrace{\Delta t(J_t \otimes \mathcal{N}_\Delta)}_{A}(\mathcal{U}) + \underbrace{(D_t \otimes I_x + \Delta t J_t \otimes \mathcal{L}_\Delta)}_{B}\mathcal{U} - \underbrace{\left(\mathbf{e}_1 \otimes \left[U^0 - \frac{\Delta t}{2} F(U^0, S^0)\right] + \Delta t(J_t \otimes I_x)\mathcal{S}\right)}_{\mathcal{C}} = 0. \tag{3.10}$$

对于 $m = 2$，方程（3.1）采用隐式、二阶、无条件稳定的 Newmark 格式进行离散 [17]:

$$\frac{U^n - 2U^{n-1} + U^{n-2}}{\Delta t^2} + \frac{1}{4}\left[F(U^n, S^n) + 2F(U^{n-1}, S^{n-1}) + F(U^{n-2}, S^{n-2})\right] = 0, \quad n = 1, 2, \ldots, N_t.$$

对于 $n = 1$， $U^{(-1)}$，在 $t = -\Delta t$处的空间解向量由方程（3.7）近似表示。完整的时空解向量 $\mathcal{U}$ 满足:

$$\underbrace{\Delta t^2 (K_t \otimes \mathcal{N}_\Delta)}_{A}(\mathcal{U}) + \underbrace{(D_{tt} \otimes I_x + \Delta t^2 K_t \otimes \mathcal{L}_\Delta)}_{B}\mathcal{U} - \underbrace{\left(\mathbf{e}_1 \otimes C^1 - \mathbf{e}_2 \otimes C^2 + \Delta t^2 (K_t \otimes I_x)\mathcal{S}\right)}_{\mathcal{C}} = 0, \tag{3.11}$$

$$C^1 = \left[2U^0 - U^{(-1)} - \frac{\Delta t^2}{2} F(U^0, S^0) - \frac{\Delta t^2}{4} F(U^{(-1)}, S^{(-1)})\right], \quad C^2 = \left[U^0 + \frac{\Delta t^2}{4} F(U^0, S^0)\right].$$

3.2. 牛顿法。我们使用牛顿法迭代求解方程（3.6）。在每次迭代 k中，解按如下方式更新:

$$\mathcal{U}_{k+1} = \mathcal{U}_k + \omega_k \mathcal{U}'_k, \tag{3.12}$$

$$J_k \mathcal{U}'_k = -f(\mathcal{U}_k), \tag{3.13}$$

其中， $\omega_k \in (0, 1]$ 为松弛因子，雅可比矩阵 J 定义如下:

$$J \equiv \frac{\partial f(\mathcal{U})}{\partial \mathcal{U}} = A'(\mathcal{U}) + B. \tag{3.14}$$

或者，我们可以求解:

$$J_k \mathcal{U}_{k+1} = J_k \mathcal{U}_k - \omega_k f(\mathcal{U}_k) := \mathcal{R}_k. \tag{3.15}$$

对于线海　rch 方法中，上述方程经过重新表述，以避免对变化的 ω_k反复求解　　:

$$J_k \widetilde{\mathcal{U}}_{k+1} = J_k \mathcal{U}_k - f(\mathcal{U}_k) := \mathcal{R}_k = A'(\mathcal{U}_k)\mathcal{U}_k - A(\mathcal{U}_k) + \mathcal{C} \tag{3.16}$$

$$\mathcal{U}_{k+1} = (1 - \omega_k)\mathcal{U}_k + \omega_k \widetilde{\mathcal{U}}_{k+1} \tag{3.17}$$

尽管方程（3.12）–（3.13）与方程（3.16）–（3.17）在数学上等价，我们更倾向于采用后者的表述。在这种形式下，量 $\mathcal{R}_k$ 作为牛顿修正方程中的修正右端项出现，因为它消去了线性项 $B\mathcal{U}_k$，仅保留非线性贡献以及初始项和边界项。在我们的数值实验中，观察到不含线性 $B\mathcal{U}_k$ 项可显著改善牛顿修正的条件性和非线性收敛行为。此外，与完整右端项 $f(\mathcal{U}_k)$ 相比，这种重新表述始终能使 $\mathcal{R}_k$ 的张量列车秩更低，从而相应降低计算成本。

对于张量列车格式下线性系统的求解（参见方程（3.13）或方程（3.16））， 我们采用双站点密度矩阵重整化群（DMRG）方法 [21] （也称为修正交替线性方案，MALS [12]）来求解张量列车格式下的线性系统。该方法具有秩自适应特性，可自动确定张量列车秩。如有需要，可根据舍入误差 $\varepsilon_{\mathrm{TT}}$ 等输入参数

和最大 TT 秩 χ对解的秩进行截断。此外，在每次双站点密度矩阵重整化群扫描中，全局牛顿系统 $J_k \mathcal{U}_{k+1} = b$ 都会化简为一系列如下形式的有效局部线性系统

$$A_{\mathrm{loc}}\, x = b_{\mathrm{loc}}, \tag{3.18}$$

其中， A_{loc} 和 b_{loc} 分别是与活动 TT 核相关的投影雅可比矩阵 J_k 和右端向量 b 。对于强非线性或对流主导问题， A_{loc} 可能会变得病态或接近秩亏，从而导致牛顿迭代停滞或不稳定。

为提高稳健性，我们在这些局部系统层面采用蒂霍诺夫正则化。具体而言，我们不求解未正则化的最小二乘问题，而是计算

$$x = \arg\min_{y} \left\{ \|A_{\mathrm{loc}} y - b_{\mathrm{loc}}\|_2^2 + \alpha \|y\|_2^2 \right\}, \tag{3.19}$$

其中， $\alpha > 0$ 是正则化参数。利用奇异值分解 $A_{\mathrm{loc}} = U \Sigma V^\top$，正则化解可显式写为

$$x = V \left(\Sigma^2 + \alpha I\right)^{-1} \Sigma U^\top b_{\mathrm{loc}} = A_{\mathrm{loc},\alpha}^{\dagger}\, b_{\mathrm{loc}}, \tag{3.20}$$

这对应于蒂霍诺夫正则化的摩尔–彭罗斯伪逆。该正则化会抑制 A_{loc}中较小奇异值的贡献，在稳定局部求解的同时保留张量列车表示的低秩结构。

算法3.1 和 3.2 概述了在这些设置下求解方程 (3.6) 的步骤。

算法 3.1 使用方程 (3.6) 和方程 (3.13) 求解非线性方程的牛顿法

输入： 初始猜测 $\mathcal{U}_0$，TT 舍入输入 ε_{TT}，χ 牛顿求解器输入 n_{iter}，ε_{res}，ε_{cor}，线搜索参数 s，n_{line}，线性系统求解器输入 $\varepsilon_{\text{DMRG}}$，$n_{sweeps}$，蒂科诺夫正则化参数 α
输出： 解 $\mathcal{U}_{k+1}$
设定 $\varepsilon_{\text{TT},k} = 10^{-3}$
步骤 1： 根据方程（ $f(\mathcal{U}_0)$ 计算初始残差（3.6），压缩并评估 $\|f(\mathcal{U}_0)\|$，$\|\mathcal{C}\|$
对于 $k = 0, 1, \ldots, n_{iter}-1$ **执行**
    步骤 2： 计算雅可比矩阵 J_k 根据方程（3.14）并压缩
    步骤 3： 使用双位点 DMRG 求解器求解线性系统方程 (3.13)，以求得修正量 $\mathcal{U}'_k$ 并压缩
        $\mathcal{U}'_k =$ 密度矩阵重整化群$(J_k,\ -f(\mathcal{U}_k)\ \alpha,\ n_{sweeps},\ \varepsilon_{\text{DMRG}},\ \chi).compress(\varepsilon_{\text{TT},\ k},\ \chi)$
    步骤 4： 通过线搜索更新解
    $\omega_k = 1$
    对于 $m = 1, \ldots, n_{line}$ **执行**
        计算 $\mathcal{U}_{k+1}$，使用方程 (3.12) 并压缩
        评估残差 $f(\mathcal{U}_{k+1})$ 的方程 (3.6)，压缩并评估 $\|f(\mathcal{U}_{k+1})\|$
        若 $\|f(\mathcal{U}_{k+1})\| < \|f(\mathcal{U}_k)\|$ **则**
            计算 $\|\mathcal{U}_{k+1}\|$
            跳出
        结束条件判断
        $\omega_k = \omega_k * s$
    结束循环
    如果 $\|f(\mathcal{U}_{k+1})\| \geq \beta\|f(\mathcal{U}_k)\|$ **则**
        $\varepsilon_{\text{TT},k} = 0.8\,\varepsilon_{\text{TT},k}$
        $\varepsilon_{\text{TT},\ k} =$ 最大值$(\varepsilon_{\text{TT},\ k},\ \varepsilon_{\text{TT}})$
    结束如果
    检查收敛性：
    如果 $\|f_{k+1}\|/\|\mathcal{C}\| < \varepsilon_{res}$ **或** $\|\mathcal{U}'_k\|/\|\mathcal{U}_{k+1}\| < \varepsilon_{cor}$ **则**
        跳出
    结束条件判断
结束循环

算法 3.2 使用方程 (3.6) 和方程 (3.15) 求解的牛顿法

输入： 初始猜测 $\mathcal{U}_0$，TT 舍入输入 $\varepsilon_{\mathrm{TT}}$，$\chi$ 牛顿求解器输入 n_{iter}，ε_{res}，ε_{cor}，线搜索参数 s，n_{line}，线性系统求解器输入 $\varepsilon_{\mathrm{DMRG}}$，$n_{sweeps}$，蒂科诺夫正则化参数 α
输出： 解 $\mathcal{U}_{k+1}$
设置 $\varepsilon_{\mathrm{TT},k} = 10^{-3}$
步骤 1： 根据方程 $f(\mathcal{U}_0)$ 计算初始残差，并根据方程 (3.6) 进行压缩和评估 $\|f(\mathcal{U}_0)\|$，$\|\mathcal{C}\|$
对于 $k = 0, 1, \ldots, n_{iter}-1$ **执行**
    步骤 2： 计算雅可比矩阵 J_k 根据方程 (3.14) 并进行压缩
    步骤 3： 计算并压缩 $\mathcal{R}_k$ 定义于方程 (3.16)
    步骤 4： 求解方程 (3.16) 的线性系统 $\widetilde{\mathcal{U}}_{k+1}$ 使用双位点 DMRG 求解器求解并压缩
    $\widetilde{\mathcal{U}}_{k+1} =$ 密度矩阵重整化群$(J_k, \mathcal{R}_k)$ $\alpha, n_{sweeps}, \varepsilon_{\mathrm{DMRG}}, \chi).compress(\varepsilon_{\mathrm{TT},k}, \chi)$
    步骤 4： 通过线搜索更新解
    $\omega_k = 1$
    对于 $m = 1, \ldots, n_{line}$ **执行**
        计算 $\mathcal{U}_{k+1}$ 使用方程 (3.17) 并压缩
        评估残差 $f(\mathcal{U}_{k+1})$（来自方程 (3.6)，压缩，并评估 $\|f(\mathcal{U}_{k+1})\|$
        如果 $\|f(\mathcal{U}_{k+1})\| < \|f(\mathcal{U}_k)\|$ **则**
            计算 $\|\mathcal{U}_{k+1}\|$
            跳出
        结束如果
        $\omega_k = \omega_k * s$
    结束循环
    如果 $\|f(\mathcal{U}_{k+1})\| \geq \beta\|f(\mathcal{U}_k)\|$ **则**
        $\varepsilon_{\mathrm{TT},k} = 0.8\,\varepsilon_{\mathrm{TT},k}$
        $\varepsilon_{\mathrm{TT},k} = \max(\varepsilon_{\mathrm{TT},k}, \varepsilon_{\mathrm{TT}})$
    结束条件判断
    检查收敛性：
    如果 $\|f_{k+1}\|/\|\mathcal{C}\| < \varepsilon_{res}$ **或** $\|\mathcal{U}_k'\|/\|\mathcal{U}_{k+1}\| < \varepsilon_{cor}$ **然后**
        跳出
    结束如果
结束循环

3.3. 压缩精度。 我们提出一种自适应压缩策略，通过将 TT 舍入误差与网格间距关联，使其与数值离散误差相匹配。采用不同阈值对张量列向量 $\mathcal{U}$、$f(\mathcal{U})$ 以及雅可比TT矩阵 J 进行压缩。

设 $\mathcal{U}_{\mathrm{exact}}$ 为连续偏微分方程的解析解，$\mathcal{U}_d$ 为离散化偏微分方程的数值解，$\mathcal{U}_{TT}$为其张量列车表示。假设 $\mathcal{U}_d$具有形式精度 $\mathcal{O}((\Delta x)^r, (\Delta t)^s)$，并定义 $C_m := \min\left(\left(\frac{\Delta x}{\Delta x_0}\right)^r, \left(\frac{\Delta t}{\Delta t_0}\right)^s\right)$，其中 Δx_0和 $\overline{\Delta t_0}$表示参考网格间距，使得参考网格产生 $C_m = 1$；我们有

$$\mathcal{E} := \mathcal{U}_{\mathrm{exact}} - \mathcal{U}_{TT} = \underbrace{\mathcal{U}_{\mathrm{exact}} - \mathcal{U}_d}_{\mathcal{E}_d} + \underbrace{\mathcal{U}_d - \mathcal{U}_{TT}}_{\mathcal{E}_r}$$

其中，$\mathcal{E}_d \sim \mathcal{O}((\Delta x)^r, (\Delta t)^s)$ 和 $\mathcal{E}_r$ 分别表示数值离散误差和 TT 舍入。

由方程 (3.13) 可得,

$$\begin{aligned}
\|\mathcal{U}_k'\| &\leq \|J_k^{-1}\|\,\|f(\mathcal{U}_k)\|,\\
\|\mathcal{U}_k'\| &\leq \sigma_{\min}^{-1}(J_k)\,\|f(\mathcal{U}_k)\|,\\
\|\mathcal{U}_k'\| &\lesssim \varepsilon_J^{-1}\,\|f(\mathcal{U}_k)\|.
\end{aligned}$$

每次牛顿迭代都会按如下方式压缩张量列变量（后缀表示相应变量）：

$$\begin{aligned}
\varepsilon_{TT} &= C_m\,\widetilde{\varepsilon}_{TT} = \mathcal{O}(\Delta t^r, \Delta x^r)\\
\varepsilon_{\mathcal{U}} &= \varepsilon_{\mathcal{U}'} = \varepsilon_J = \varepsilon_{TT}\\
\varepsilon_f &= \varepsilon_{\mathcal{R}} = \sigma_{\min}(J)\,\varepsilon_r \approx \varepsilon_J\varepsilon_r \approx 0.25\varepsilon_r.
\end{aligned}$$

3.4. 多层策略。 我们提出一种多层方法，利用由粗到细的计算网格层级来加速收敛，并提高牛顿法的稳健性。空间和时间上相邻两个层级的网格间距之比（细化比）均取为 2。

在此方法中，首先在粗网格上近似求解非线性问题，此处的计算成本较低。随后将粗网格解延拓（插值）到更细网格，作为该层级牛顿求解器的初始猜测。此过程在多个层级上递归进行，由粗到细；每个更细层级上的牛顿求解都能受益于信息逐步增多的初始猜测。

通过将网格细化与牛顿法相结合，该策略降低了对较差初始猜测的敏感性，增强了收敛性；与从头开始直接在最细网格上求解非线性问题相比，还能显著降低计算成本。

考虑由以下各层组成的层级结构： n_l 时空网格，$\Omega^{(l)}$，其中 $l = 0,\ 1,\ \ldots\ldots,\ n_l - 1$，细化比为二。每一层 l 的网格尺寸为 $(\Delta x,\ \Delta t)2^l$，其中 Δx 和 Δt 分别为最细层级 ($l = 0$) 上的空间和时间步长。由 $\Omega^{(l)}$ 导出的离散非线性方程组方程 (3.6) 为:

$$f^{(l)} = f^{(l)}(\mathcal{U}^{(l)}) = 0,$$

其中，$f^{(l)}:\ \mathbb{R}^{N^{(l)}} \to \mathbb{R}^{N^{(l)}}$ 是网格 Ω 上的非线性算子[(l)]，$\mathcal{U}^{(l)} \in \mathbb{R}^{N^{(l)}}$ 是 Ω 上的时空解向量[(l)]，$N^{(l)} = (N_x/2^l) \times (N_t/2^l)$ 是 Ω 上的自由度数量[(l)]。算法 3.3 描述了多层牛顿法的步骤。

算法 3.3 方程 (3.6) 的多层时空求解器

1：输入：初始条件 $U^0(x)$**；层级数** n_ℓ**；延拓算子** $\mathcal{P}^{(\ell,\ell+1)}$**；张量列车参数** $\varepsilon_{\mathrm{TT}}, \chi$**；牛顿参数** $n_{\mathrm{iter}}, \varepsilon_{\mathrm{res}}, \varepsilon_{\mathrm{cor}}, s, n_{\mathrm{line}}$**；密度矩阵重整化群参数** $\varepsilon_{\mathrm{DMRG}}, n_{\mathrm{sweeps}}$**；蒂科诺夫正则化参数** α**2：输出：最细层级上的时空解** $\mathcal{U}^{(0)}$ **3：对于** $\ell = n_\ell - 1, \ldots, 0$**，执行** ▷ **从最粗层级** $\ell = n_\ell - 1$ **到最细层级** $\ell = 0$**4：在** $\mathbf{\Omega}^{(\ell)}$ **上采样初始条件，并构造张量列车向量** $(U^0)^{(\ell)}$**5：组装张量列车算子** $A^{(\ell)}, B^{(\ell)}, \mathcal{C}^{(\ell)}$**6：如果** $\ell = n_\ell - 1$ **，则** ▷ **最粗层级 7：构造初始猜测：** $\mathcal{U}_0^{(\ell)} = \mathbf{1}_{n_t^{(\ell)}} \otimes (U^0)^{(\ell)}$**，其中** $\mathbf{1}_{n_t^{(\ell)}} = (1, \ldots, 1)^T \in \mathbb{R}^{n_t^{(\ell)}}$ **8：否则 9：从前一层级延拓解：** $\mathcal{U}_0^{(\ell)} = \mathcal{P}^{(\ell,\ell+1)}\mathcal{U}^{(\ell+1)}$**10：结束如果 11：在层级** ℓ**上，使用算法 3.1 或算法 3.2，以初始猜测** $\mathcal{U}_0^{(\ell)}$ **求解非线性问题方程 (3.6) 12：存储收敛解** $\mathcal{U}^{(\ell)}$**13：结束循环 14：返回** $\mathcal{U}^{(0)}$

使用延拓算子 P 将解从粗网格（第 l 层）传递到更细网格（第 $l-1$ 层），从而细化近似解。给定粗网格解 $\mathcal{U}^{(l)}$，延拓过程表示为 $\mathcal{U}^{(l-1)} = P^{(l-1,l)}\mathcal{U}^{(l)}$，其中 $\mathcal{U}^{(l-1)}$ 为插值细网格解。在二维情况下采用双线性插值，通过相邻粗网格值的加权平均计算细网格值（参见 [16] 和 [27] 的第 68–70 页）。时空延拓算子的张量列车表示由空间方向 (P_x) 和时间方向 (P_t) 上的一维线性延拓算子 [14] 的克罗内克积得到，如下：

$$P^{(l-1,l)} = P_t^{(l-1,l)} \otimes P_x^{(l-1,l)} = P_t^{(l-1,l)} \bowtie P_x^{(l-1,l)}. \tag{3.21}$$

3.5. 计算复杂度。 在以下分析中，我们假设这三种方法均采用隐式欧拉时间离散和空间导数的二阶中心差分。设空间大小为 N_x、时间大小为 N_t，其中 $q_x = \log_2 N_x$、$q_t = \log_2 N_t$，并令 $q = q_x + q_t$。令 k_N 为牛顿迭代次数。对于 QTT，记向量 / 迭代量的典型秩为 r，并令 $\chi = \max(r)$ 为键维数，即所有核中解的最大秩。

3.5.1. 经典时间步进。在经典时间步进方法中，我们在每个时间步长采用牛顿法，并使用稠密直接求解器求解每次牛顿迭代对应的线性系统。每次牛顿迭代都要求解一个含有 N_x 个未知数的稠密线性系统。假设每个时间步长需要进行 k_N 次牛顿迭代，则 N_t 个时间步长的计算成本为：

$$\text{Work} \;=\; \mathcal{O}\big(k_N\, N_t\, N_x^3\big), \qquad \text{Memory} \;=\; \mathcal{O}(N_x^2).$$

3.5.2. 单层时空 QTT。时空解向量 $\mathcal{U} \in \mathbb{R}^{N_x N_t}$ 以 QTT 表示，包含 $q = q_x + q_t = \log_b(N_x N_t)$ 个二进制模态（每个模态的大小为 $b = 2$）。每次扫描中，DMRG 方法针对每个位置求解一个包含 br^2 个未知数的方形稠密线性系统

为此，我们采用基于 SVD 的最小二乘法（复杂度按立方增长），其计算量为 $\mathcal{O}(b^3 r^6)$。若秩 r 在所有 DMRG 迭代中均由 χ 统一有界，则牛顿法中的一次基于 DMRG 的线性求解，经过 k_D 次扫描 的计算量（忽略常数因子）为

$$\text{Work per Newton step} \;=\; \mathcal{O}\big(k_D q\,\chi^6\big), \qquad \text{Memory} \;=\; \mathcal{O}\big(q\,\chi^2\big).$$

总计：

$$\text{Work} \;=\; \mathcal{O}\big(k_N\,k_D\,q\,\chi^6\big) \;=\; \mathcal{O}\big(k_N\,k_D\,\log_2(N_x\,N_t)\,\chi^6\big).$$

3.5.3. 多层时空 QTT。假设每个层级 ℓ =0，1，.....，$L-1$ 上各进行 k_N 次牛顿迭代，且每个层级的网格大小为 $(2^{q_x-l})\times(2^{q_t-l})$，因此每个层级有 $q^{(\ell)}=q-2\ell$ 个核，其中 q 是最细层级的核数。通常，$L=\min(q_t,\ q_x)-2$，其中 $q_t,\ q_x\geq 2$。令 $\chi^{(\ell)}$ 表示层级 ℓ 上的最大 QTT 秩。每次牛顿迭代使用 k_D 次 DMRG 扫描。那么总计算量（忽略常数因子）为

$$\text{Work} \;=\; \sum_{\ell=0}^{L-1} k_N\,k_D\,\mathcal{O}\big(q^{(\ell)}(\chi^{(\ell)})^6\big) = \sum_{\ell=0}^{L-1} k_N\,k_D\,\mathcal{O}\big((q-2\ell)(\chi^{(\ell)})^6\big)$$

峰值内存则由最细层级主导：

$$\text{Memory} \;=\; \mathcal{O}\big(q\,(\chi^{(0)})^2\big).$$

如果各层级的秩单调不减，且统一有界于 χ，并且每次细化都会使 $q^{(\ell)}$ 增加 2，则

$$\begin{aligned}\text{Work} \;&=\; \mathcal{O}\big(\,k_N\,k_D\,(q-L+1)L\,\chi^6\big) \;=\; \mathcal{O}\big(\,k_N\,k_D\,(\log_2(N_x\,N_t)-L+1)L\,\chi^6\big),\\ \text{Memory} \;&=\; \mathcal{O}\big(q\,\chi^2\big).\end{aligned}$$

注意，二进制物理维数带来的常数 b 被吸收到大-$\mathcal{O}$ 项中；多层策略严格按照从粗到细的顺序进行，不会来回循环。

4. 非线性偏微分方程的数值实验。接下来，我们在非线性初边值问题上比较多层时空 TT（ML）、单层时空 TT（SL）与经典时间步进（CT）方法，三者采用相同的数值离散格式。数值误差（e）与相对误差（e_r）通过 L^2- 范数进行量化，其定义为，

$$\|e\|_{L^2} = \left(\int_\Omega |e_j|^2\,d\Omega\right)^{1/2} \approx \left(\sum_{j=1}^{N_x} |e_j|^2\Delta x\right)^{1/2}, \quad \|e_r\| \coloneqq \frac{\|e\|_{L^2}}{\|u\|_{L^2}} \tag{4.1}$$

除非另有说明，时间导数采用一阶隐式欧拉离散，空间导数采用二阶中心差分离散。以下测试中所用全部离散算子的张量列车表示见附录A。

解的精度、收敛性和运行时间取决于多个参数，包括空间和时间步长（Δx、Δt）、离散化的形式精度、舍入误差 $\varepsilon_{\mathrm{TT}}$ 和$\varepsilon_{\mathrm{DMRG}}$、收敛准则、最大 TT 秩 χ以及网格层级数量。截断误差 $\varepsilon_{\mathrm{TT}}$ 和 χ 应小于离散误差。通常，这些量无法事先获知，必须通过经验进行估计。

下文将展示我们的方法对于各种类型非线性偏微分方程的收敛性的数值证据。值得注意的是，在数值求解中，对正则化的需求主要出现在非线性双曲型问题中，此时整体式时空系统的雅可比矩阵会变得严重病态。就物理意义而言，双曲动力学的特征包括对流输运、陡峭梯度，以及激波或波前等尖锐结构的形成。在全局时空离散中，这些特征会诱发强烈的各向异性，导致雅可比矩阵的奇异值谱快速衰减。尤其是，随着系统规模增大，病态性会进一步加剧。

在张量列车框架中，这种病态性体现为局部双站点密度矩阵重整化群问题。将全局雅可比矩阵投影到低秩张量列车流形上得到的有效局部矩阵 A_{loc}，对于刚性非线性问题（例如通过 KdV 方程和正弦-戈登方程模拟的孤子传播问题）可能会变得近乎秩亏。随着系统规模增大，这些影响会更加明显。因此， A_{loc}中会出现较小的奇异值，使相应的最小二乘问题变得不适定，并导致牛顿更新放大数值噪声，而非减小残差。

这种行为与抛物型问题中观察到的情况有本质区别；在抛物型问题中，扩散会对算子产生内在的正则化作用。在这类问题中，雅可比矩阵在网格细化过程中始终保持良好的条件性，张量列车秩保持适中，局部密度矩阵重整化群系统无需额外稳定化便具有良好的适定性。然而，对于双曲型问题，强非线性与尖锐时空特征相结合，因此必须进行显式正则化，以稳定局部求解并确保牛顿收敛可靠。

所有数值实验均使用 Python 实现（Python 3.12.0），并在一台 2015 款 MacBook Pro 上运行。该设备配备 2.5 GHz Intel Core i7 处理器和 16 GB、1600 MHz 的 DDR3 内存。

4.1. 非线性抛物型偏微分方程：Fisher-KPP 方程。费希尔-科尔莫戈罗 v--彼得罗夫斯基-皮斯库诺夫（Fisher-KPP）方程是反应-扩散方程的一个典型例子。费希尔最初提出该方程 [10] ，用于模拟有利等位基因在群体遗传学中的空间传播。如今，该方程已成为多个领域的基础，包括生态学（物种入侵动力学）、燃烧理论（火焰前沿传播）、流行病学（疾病传播）和伤口愈合过程。

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2} + ru(1-u), \quad x \in [-L/2, L/2], \quad t > 0 \tag{4.2}$$

其中， $u(x,\ t)$ 表示种群密度或浓度， $D > 0$ 表示扩散系数， $r > 0$ 表示内禀增长率。线性算子和非线性算子分别为 $\mathcal{L}_\Delta = -D\,\frac{\partial^2}{\partial x^2} - r$ 和 $\mathcal{N}_\Delta(u) = ru^2$。空间导数采用二阶中心差分格式离散，时间离散则采用隐式欧拉时间离散。

该方程存在行波解 $u(x,\ t) = U(z)$，其中 $z = x - ct$ 连接不稳定态 $u = 0$ 与稳定态 $u = 1$，满足 $\lim_{z\to+\infty} U(z) = 0$ 且 $\lim_{z\to-\infty} U(z) = 1$。虽然不存在一般的闭式解，但 Ablowitz 和 Zeppetella [1] 针对特殊情形 $c = 5/\sqrt{6}$ （其中 $D = r = 1$）找到了精确解析解：

$$U(z) = \frac{1}{\left[1 + Ae^{z/\sqrt{6}}\right]^2} \tag{4.3}$$

其中 $A > 0$ 决定波的位置。我们选择 $A = 1$, $L = 40$, 初始条件取自方程 (4.3), 其中 $t = 0$, 边界条件为 $u(-L/2,\ t) = 1$, $u(L/2,\ t) = 0$ 并计算相对于解析解的数值误差。

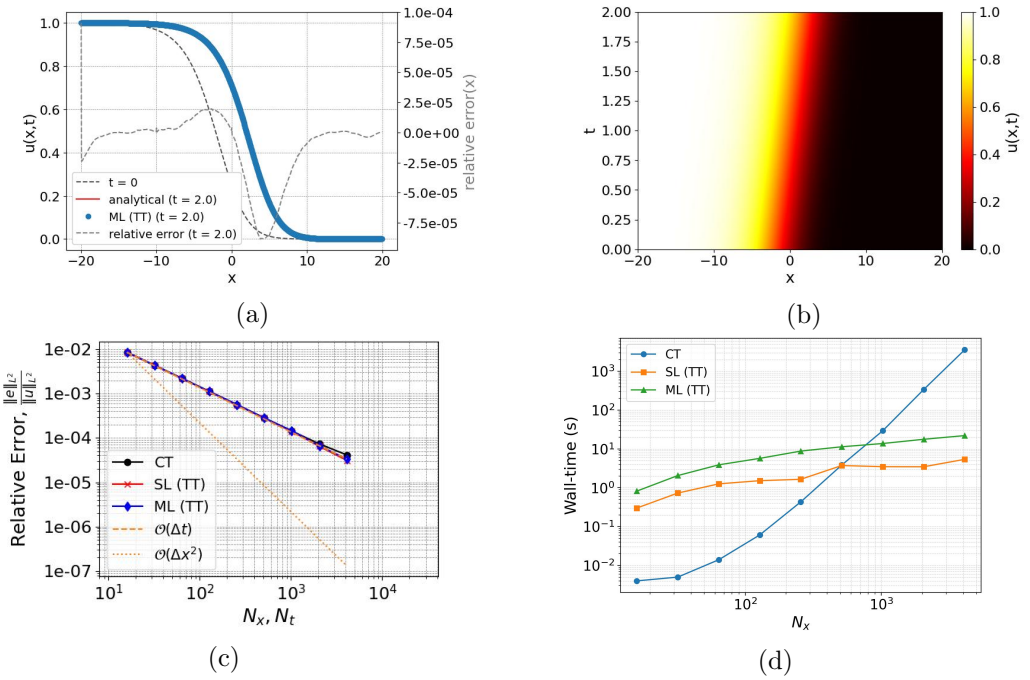


图 1: 费希尔-KPP 方程 方程 (4.2) 的行波解: $D = 1$, $r = 1$, $N_x = N_t = 2^{10}$, $\Delta t = 2/N_t$; (a) 与 $t = 2$ 时的解析解对比; (b) 时空解的等高线图; (c) 数值误差 ($\mathcal{O}(\Delta t,\ \Delta x^2)$) 随网格分辨率 N_x, N_t 的收敛情况; (d) 经典时间步进方法 (CT) 与单层 (SL) 和多层 (ML) 时空方法的运行时间对比。

图 1(a) 显示了多层 TT 方法在 $t = 2$ 、网格为 $N_x = N_t = 2^{10}$ 和 $\Delta t = 2/N_t$ 时得到的数值解与解析解高度吻合。图1(b) 展示了 $(x,\ t)$ 平面上的解。图 1(c) 显示, 在 $t = 2$时, 数值误差按 $\mathcal{O}(\Delta t,\ \Delta x^2)$ 收敛, 这与时间方向采用隐式欧拉格式、空间方向采用二阶中心差分的结果一致; 两者的形式精度分别为 $\mathcal{O}(\Delta t)$ 和 $\mathcal{O}(\Delta x^2)$。

表2 对三种方法（经典时间步进、单层和多层）的误差收敛和计算性能进行了定量比较。结果表明, 单层（SL）和多层（ML）张量列车时空方法所达到的误差几乎与经典时间步进（CT）基准相同。这种高度一致性证实, TT 舍入误差（由ε_{TT}控制）低于空间和时间离散误差的水平, 从而确保低秩近似不会影响解的精度。

多层方法的一个关键优势是非线性收敛性能更好。如表2所示, 与单层方法相比, ML方法始终需要更少的牛顿迭代即可收敛, 在更细网格上尤为明显。这突显了粗网格初始化的有效性

$N_x \times N_t$	牛顿迭代		运行时间（秒）			最大秩 χ		相对误差 $\|e\|_{L^2}/\|u\|_{L^2}$			$\|e\|_{L^2}$
	SL	ML	SL	ML	CT	SL	ML	SL	ML	CT	CT
$2^4 \times 2^4$	3	4	0.30	0.82	0.004	8	8	8.51e-03	8.51e-03	8.51e-03	3.77e-02
$2^5 \times 2^5$	4	4	0.73	2.06	0.005	9	9	4.32e-03	4.32e-03	4.32e-03	1.91e-02
$2^6 \times 2^6$	4	4	1.26	3.89	0.014	9	9	2.19e-03	2.19e-03	2.19e-03	9.71e-03
$2^7 \times 2^7$	4	3	1.52	5.71	0.061	9	9	1.11e-03	1.11e-03	1.11e-03	4.91e-03
$2^8 \times 2^8$	4	4	1.65	8.79	0.432	9	9	5.59e-04	5.64e-04	5.57e-04	2.47e-03
$2^9 \times 2^9$	7	4	3.73	11.35	3.841	9	9	2.80e-04	2.80e-04	2.80e-04	1.24e-03
$2^{10} \times 2^{10}$	6	4	3.49	13.78	29.329	9	9	1.41e-04	1.45e-04	1.42e-04	6.27e-04
$2^{11} \times 2^{11}$	6	7	3.47	17.70	338.445	9	9	6.47e-05	6.63e-05	7.33e-05	3.24e-04
$2^{12} \times 2^{12}$	9	8	5.39	22.02	3570.864	9	9	3.12e-05	3.34e-05	4.12e-05	1.83e-04

表 2：采用 $D = 1$ 和 $r = 1$求解 Fisher–KPP方程的数值解。张量列车求解器的参数为 $\varepsilon_{\mathrm{TT}} = 10^{-6}$、$\varepsilon_{\mathrm{DMRG}} = 10^{-3}$ 和 $\varepsilon_{\mathrm{Newton}} = 10^{-5}$。DMRG扫描的最大次数固定为 3。多层方法采用 $n_l = \min(q_x,\ q_t) - 1$ 个层级。相对误差定义为 $\|e\|_{L^2}/\|u\|_{L^2}$，其中 $\|u(x,\ t = 2)\|_{L^2} = 4.43$。最后一列报告了经典时间步进（CT）解的绝对误差。SL 表示单层张量列车方法，ML 表示多层张量列车方法，CT 表示经典时间步进。

在提供高质量初始猜测方面的策略，从而增强牛顿求解器的鲁棒性。

图 1 (d) 清晰地展示了整体张量列车方法的计算效率。随着网格分辨率提高，经典时间步进方法的运行时间呈现陡峭的指数式增长，这直接源于其 $\mathcal{O}(N_t N_x^3)$ 缩放特性。与之形成鲜明对比的是，单层和多层方法利用量化张量列车（QTT）格式的对数级复杂度，使运行时间呈现出显著更优的对数线性增长。这凸显了张量列车框架的主要计算优势：它缓解了整体式时空方法固有的维数灾难，使大规模、高分辨率模拟切实可行。

4.2. 非线性抛物型偏微分方程：粘性伯格斯方程。考虑以下非线性粘性伯格斯方程：

$$u_t + \frac{1}{2}(u^2)_x = \nu u_{xx}, \quad x \in [0,1], \quad t \in [0,1]. \tag{4.4}$$

线性算子和非线性算子 $\mathcal{L}_\Delta = -\nu \frac{\partial^2}{\partial x^2}$，以及 $\mathcal{N}_\Delta(u) = \frac{1}{2}\frac{\partial}{\partial x}(u^2)$，分别采用二阶中心差分格式进行离散。我们考虑以下解析解[28]，以验证数值方法并评估其准确性和性能，方程（4.4）。

$$u(x,t) = \frac{2\nu\pi \exp\{(-\pi^2\nu t)\}\sin(\pi x)}{a + \exp\{(-\pi^2\nu t)\}\cos(\pi x)}, \qquad a > 1, \quad x \in [0,1].$$

初始条件取为 $t = 0$ 处的精确解，并对所有 t 施加齐次狄利克雷边界条件 $u(0,t) = u(1,t) = 0$。时间步长选为$\Delta t = 1/N_t$。

我们将雷诺数定义为 $Re = u_{\max} L/\nu = 2\pi L/a$，其中 L 表示空间区域的长度，$u_{\max} = 2\nu\pi/a$ 是最大平流速度。由于 $a > 1$，最大

对于上述初始条件，可达到的雷诺数为 $Re_{\max} = 2\pi L = 2\pi$，即 $\mathcal{O}(1)$。这表明扩散效应显著，且控制偏微分方程主要处于抛物型区间。

相比之下，双曲型区域需要 $Re \gg 1$，而在当前的初始条件选择下无法实现，因为雷诺数直接受到黏度的限制。因此，为了研究双曲型区域，我们在第 4.3节考虑另一类初始条件，这类条件允许较大的雷诺数，并会导致激波形成。

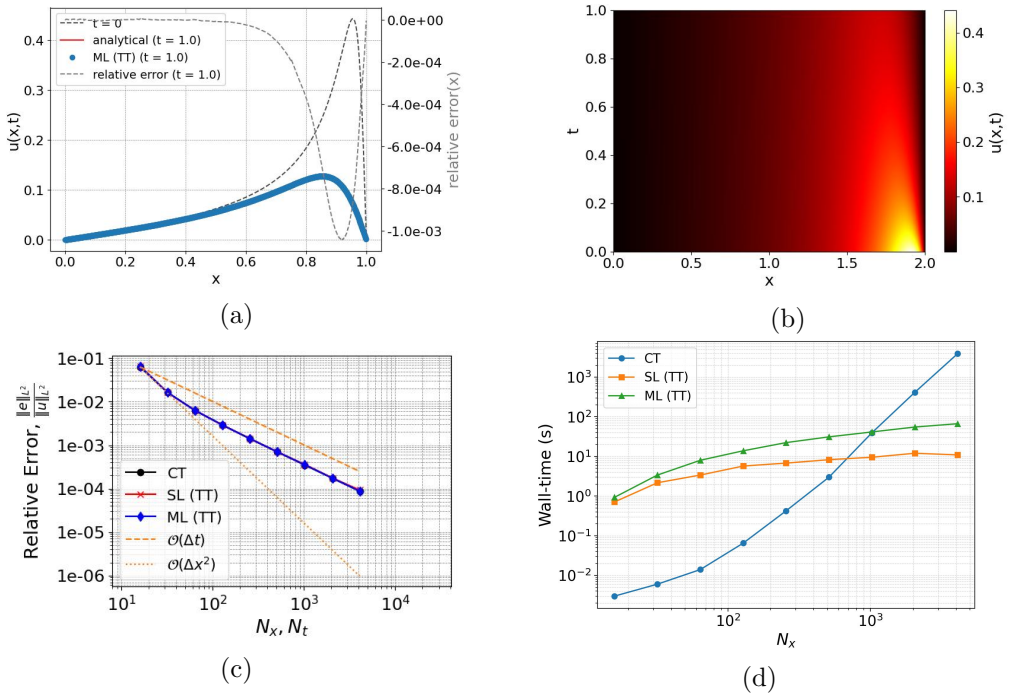


图 2：抛物型区间内非线性黏性 Burgers 方程方程（4.4）的数值解：$\nu = 10^{-2}$，$a = 1.01$，$N_x = N_t = 2^{10}$，$\Delta t = 1/N_t$，(a) 在 $t = 1$ s处与解析解的比较，(b) $xt-$ 平面上的时空解，(c) 数值误差收敛情况（$\mathcal{O}(\Delta t,\ \Delta x^2)$）与网格分辨率 N_x，N_t的关系，以及 (d) 经典时间步进方法（CT）、单层（SL）和多层（ML）时空方法的运行时间比较。

考虑到 $a = 1.01$ 和 $\nu = 0.01$，图 2(a) 展示了数值解（$N_x = N_t = 2^{10}$ 和 $\Delta t = 1/N_t$）与解析解在 $t = 1$处高度吻合。图 2(b) 展示了（x，t）平面上的解。图 2(c) 显示，在 $t = 1$ 处的数值误差按 $\mathcal{O}(\Delta t,\ \Delta x^2)$ 收敛，这与时间方向采用隐式欧拉格式、空间方向采用二阶中心差分的结果一致，两者的形式精度分别为 $\mathcal{O}(\Delta t)$ 和 $\mathcal{O}(\Delta x^2)$。

表3总结了抛物型Burgers方程的计算表现比较。基于张量列车的求解器（SL和ML）保留了经典（CT）方法的离散精度，因为它们各自的误差几乎完全相同。这表明，对于所选容差，张量压缩不会损失精度。多层策略

$N_x \times N_t$	牛顿迭代		运行时间（秒）			最大秩 χ		相对误差 $\|e\|_{L^2}/\|u\|_{L^2}$			$\|e\|_{L^2}$
$2^{q_x} \times 2^{q_t}$	单层	ML	SL	ML	CT	SL	ML	SL	ML	CT	CT
$2^4 \times 2^4$	6	4	0.71	0.93	0.003	7	7	6.59e-02	6.59e-02	6.34e-02	4.65e-03
$2^5 \times 2^5$	6	5	2.17	3.39	0.006	9	9	1.64e-02	1.64e-02	1.63e-02	1.16e-03
$2^6 \times 2^6$	6	5	3.39	8.00	0.014	10	10	6.33e-03	6.32e-03	6.30e-03	4.47e-04
$2^7 \times 2^7$	7	4	5.72	13.88	0.065	10	10	2.91e-03	2.91e-03	2.90e-03	2.05e-04
$2^8 \times 2^8$	7	5	6.79	22.32	0.424	10	10	1.42e-03	1.41e-03	1.41e-03	9.98e-05
$2^9 \times 2^9$	7	4	8.27	31.18	2.998	10	10	7.00e-04	7.01e-04	7.00e-04	4.94e-05
$2^{10} \times 2^{10}$	7	4	9.54	41.47	39.368	10	10	3.61e-04	3.51e-04	3.49e-04	2.81e-05
$2^{11} \times 2^{11}$	8	5	12.03	55.15	412.577	10	10	1.76e-04	1.75e-04	1.74e-04	1.23e-05
$2^{12} \times 2^{12}$	7	5	10.93	66.86	3912.819	10	10	9.26e-05	8.71e-05	8.70e-05	6.14e-06

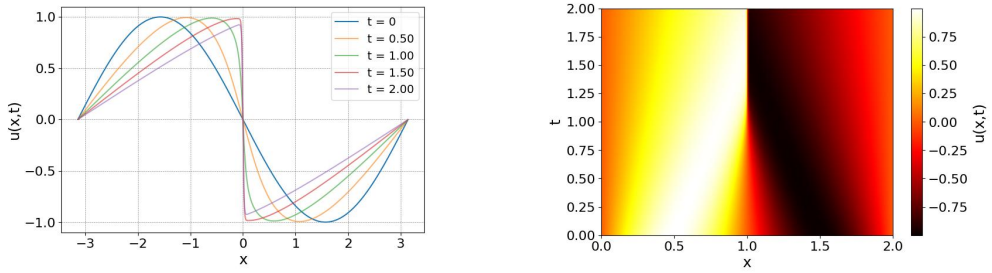
表 3：使用 $\nu = 0.01$ 和 $a = 1.01$求解Burgers方程的数值解。张量列车求解器的参数为 $\varepsilon_{\mathrm{TT}} = 10^{-6}$、$\varepsilon_{\mathrm{DMRG}} = 10^{-3}$和 $\varepsilon_{\mathrm{Newton}} = 10^{-5}$。最大DMRG扫描次数固定为3，多层方法采用 $n_l = \min(q_x, q_t) - 1$个层级。相对误差定义为 $\|e\|_{L^2}/\|u\|_{L^2}$，其中 $\|u(x, t = 1)\|_{L^2} = 7.06 \times 10^{-2}$。最后一列报告经典时间步进（CT）解的绝对误差。SL表示单层张量列车方法，ML表示多层张量列车方法，CT表示经典时间步进。

（ML）在非线性求解器效率方面具有明显优势，始终以更少的迭代次数收敛。然而，最显著的差异在于计算扩展性。图2 （d）表明，CT方法的运行时间随网格细化呈指数级激增，而借助压缩的QTT 表示，单层方法和ML方法的运行时间则呈现出远为可控的对数线性增长。

4.3. 非线性双曲型偏微分方程：黏性伯格斯方程——激波形成。 由方程（4.4）给出的黏性伯格斯方程是一个基本的非线性偏微分方程，同时表现出对流效应与扩散效应，其中 $\nu > 0$ 为运动黏度。对于初始条件 $u_0(x) = -\sin(x)$在定义域 $-\pi \le x \le \pi$上，非线性对流项会使解变陡，从而导致激波形成，此时解的梯度变得极为陡峭。在无黏极限($\nu \to 0$)下，解会出现真正的间断，形成满足Rankine-Hugoniot跳跃条件的激波。对于有限黏度，激波会在厚度为 $O(\nu)$的边界层内被平滑，从而形成陡峭但连续的过渡。

方程（4.4）在 $\nu = 10^{-2}$ 上求解，定义域为 $x \in [-\pi, \pi]$，并以网格尺寸 $N_x = N_t = 2^{10}$ 进行离散，$\Delta t = 1/N_t$。图 3（a）展示了初始正弦波演变为具有陡峭梯度的锯齿状剖面的过程，说明了非线性陡化与黏性扩散之间的竞争。这种竞争是 Burgers方程的典型特征，而该方程是用于描述激波物理、流体动力学和交通流的简化模型。图 3（b）展示了时空解，其中激波由 $t = 1$处形成的一条陡峭特征线标示。

4.4. 非线性双曲型偏微分方程：正弦-戈登方程。 正弦-戈登方程是一种非线性双曲型偏微分方程，出现在微分几何、场论和非线性波动力学中，可作为评估精度与稳定性的具有挑战性基准，对象为数值



(a) (b)

图 3：通过求解非线性黏性 Burgers 方程 方程（4.4），模拟正弦波演变为激波的过程： $\nu = 10^{-2}$, $N_x = N_t = 2^{10}$, $\Delta t = 1/N_t$, （a）每隔 10 个时间步长的解；（b）时空解的等值线。

方法，用于非线性波动方程。

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} + \sin u = 0, \tag{4.5}$$

方程（4.5）采用隐式欧拉格式进行时间离散。线性算子 $\mathcal{L}_\Delta = -\frac{\partial^2}{\partial x^2}$ 采用二阶中心有限差分格式进行离散，非线性算子$\mathcal{N}_\Delta(u) = \sin(u)$ 通过交叉逼近 [19]以张量列车格式进行评估。初始条件 $u(x, 0)$ 和$\frac{\partial u}{\partial t}(x, 0)$ 由解析扭结孤子解 [5] 在区域 $x \in [a, b]$内导出，其中 $a = -10$, $b = 15$:

$$u(x,t) = 4\arctan\left(\exp\left(\gamma(x - ct - x_0)\right)\right),$$
$$u(a,t) = 0, \quad \frac{\partial u}{\partial x}(b,t) = 0,$$

其中 $\gamma = \frac{1}{\sqrt{1-c^2}}$ 是洛伦兹因子， c是行波速度或孤子的速度 n.

图 4 展示了使用隐式欧拉时间离散，在空间–时间网格上获得的正弦–戈登方程数值解，其中 $N_x = 2^{10}$, $N_t = 2^8$，时间步长为 $\Delta t = 10/N_t$。结果表明，在整个模拟区间内，数值近似与解析解高度吻合。

图 (a) 对比了最终时刻 $t = 10$ 的数值解与解析解。两条曲线几乎无法区分，证实该数值方法准确地捕捉了扭结解的形状和传播。图 (b) 展示了数值解的空间–时间等值线图，呈现出平滑的时间演化和稳定的孤子传播，且没有伪振荡。

图 (c) 展示了数值误差随网格细化的收敛情况。与隐式欧拉离散一致，误差相对于时间步长呈一阶收敛， $\mathcal{O}(\Delta t)$。当空间网格足够精细时，时间离散误差占主导，这解释了观察到的收敛行为。

面板 (d) 比较了经典时间步进方法 (CT) 与单层 (SL) 和多层 (ML) 时空求解器的实际运行时间。SL 和 ML 方法的

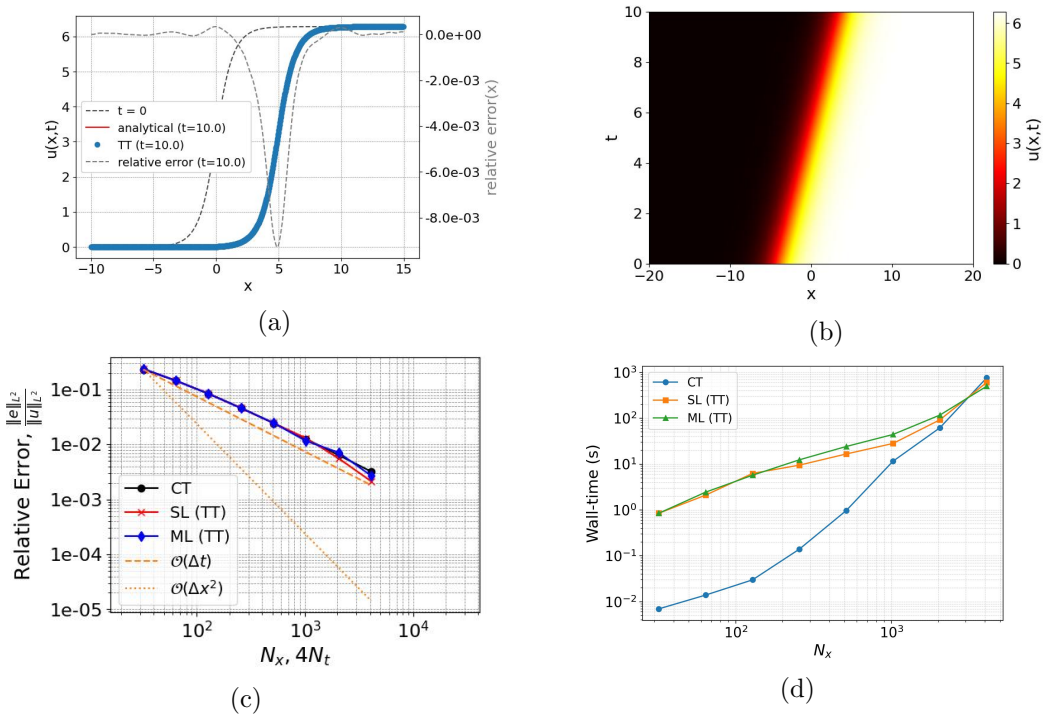


图 4：正弦–戈登方程的数值解（4.5）。离散化采用 $N_x = 2^{10}$和 $N_t = 2^8$，时间步长为 $\Delta t = 10/N_t$。（a） $t = 10$处的数值解和解析解，（b）数值解的时空等高线图，（c）网格细化下数值误差的收敛性，展示了一阶时间精度，$\mathcal{O}(\Delta t)$，这是由于采用了隐式欧拉时间离散；（d）经典时间步进方法（CT）与单层（SL）和多层（ML）时空方法的实际运行时间。

计算成本在所有测试分辨率下都非常相近。这表明，多层级结构带来的额外工作，实际上被粗网格初始化所减少的牛顿迭代次数抵消，因此 SL 和 ML 的总体运行时间相当。对于所考虑的最大问题规模（对应 12 个量子比特），SL 和 ML 的表现都优于经典时间步进方法。这一现象反映了这些方法在复杂度上的根本差异：SL 和 ML 时空求解器的计算量随时空网格大小呈对数线性增长，而经典时间步进方法的计算成本则随问题规模呈指数增长。

在最高网格分辨率下，基于张量列车的求解器的计算成本增长得更为明显。这主要是由于 TT 键维数有所增加，而这会直接影响密度矩阵重整化群求解器的成本。具体而言，密度矩阵重整化群局部求解的复杂度按 $\mathcal{O}(\chi^6)$ 缩放，因此，即使键维数仅适度增加，也会导致运行时间明显上升。尽管如此，在本研究考察的分辨率下，基于张量列车的求解器仍具有竞争力。预计通过开发更

$N_x \times N_t$	牛顿迭代		运行时间（秒）			最大秩 χ		相对误差 $\|e\|_{L^2}/\|u\|_{L^2}$			$\|e\|_{L^2}$	α
	SL	ML	SL	ML	CT	SL	ML	SL	ML	CT	CT	
$2^5 \times 2^3$	6	6	0.85	0.85	0.007	6	6	2.39e-01	2.39e-01	2.33e-01	4.45e+00	10^{-6}
$2^6 \times 2^4$	7	4	2.11	2.45	0.014	8	8	1.46e-01	1.46e-01	1.45e-01	2.78e+00	10^{-6}
$2^7 \times 2^5$	8	3	6.26	5.72	0.03	9	9	8.41e-02	8.39e-02	8.46e-02	1.62e+00	10^{-6}
$2^8 \times 2^6$	10	4	9.49	12.41	0.14	11	11	4.61e-02	4.61e-02	4.64e-02	8.88e-01	10^{-7}
$2^9 \times 2^7$	10	6	16.61	24.19	0.97	13	13	2.46e-02	2.47e-02	2.45e-02	4.68e-01	10^{-7}
$2^{10} \times 2^8$	11	6	28.19	44.27	11.56	17	14	1.30e-02	1.16e-02	1.26e-02	2.40e-01	10^{-8}
$2^{11} \times 2^9$	30	11	92.24	117.07	61.81	15	15	5.61e-03	7.15e-03	6.41e-03	1.22e-01	10^{-8}
$2^{12} \times 2^{10}$	14	12	603.25	498.24	761.40	17	17	2.14e-03	2.70e-03	3.22e-03	6.16e-02	10^{-8}

表 4：正弦–戈登扭结孤子解的数值结果。张量列车求解器的参数为 $\varepsilon_{\mathrm{TT}} = 10^{-4}$, $\varepsilon_{\mathrm{DMRG}} = 10^{-3}$ 和 $\varepsilon_{\mathrm{Newton}} = 5 \times 10^{-4}$。密度矩阵重整化群扫描的最大次数固定为 2，多层方法采用 $n_l = \min(q_x, q_t) - 2$ 个层级。SL 表示单层张量列车方法，ML 表示多层张量列车方法，CT 表示经典时间步进。蒂科诺夫正则化参数记为 α。在 $t = 10$ 处评估误差，并使用$\|u\|_{L^2} = 19.1$进行归一化。

高效的局部求解器以及预条件策略的引入，可以大幅减轻对键维数的依赖，并将良好的缩放特性扩展到更大规模的问题。

4.5. 非线性色散偏微分方程：科特韦格–德弗里斯（KdV）方程。科特韦格–德弗里斯（KdV）方程是一种基本的非线性偏微分方程，用于描述浅水波和非线性色散波的演化。

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0, \quad x \in [-L/2, L/2]. \tag{4.6}$$

它以方程（4.6）的形式结合了非线性与色散，由此产生孤子——能够在随时间演化及相互作用过程中保持形状不变的局域波包。由于三阶导数和非线性对流项带来的刚性，KdV方程给数值模拟带来了显著挑战。要准确捕捉孤子动力学，必须谨慎处理色散与非线性，因此 KdV 是验证非线性求解器的理想测试平台。

方程（4.6）采用隐式欧拉时间步进格式进行离散，线性算子$\mathcal{L}_\Delta = \frac{\partial}{\partial x^3}$和非线性算子 $\mathcal{N}_\Delta(u) = u\frac{\partial u}{\partial x} \equiv \frac{1}{2}\frac{\partial u^2}{\partial x}$（改写为守恒形式）采用二阶中心差分格式进行离散。

我们考虑长度为 $L = 30$ 的空间区域，并施加齐次狄利克雷边界条件。初始条件由解析解 [11, 9]（如下所示）给出，该解对应于速度为 $c = 1$、中心位于 $x_0 = -1$的孤子。该解满足 $\lim_{x\to\pm\infty} u(x,\ t) = 0$，同时也被用作量化数值误差的参考。

$$u(x,t) = 3c\,\mathrm{sech}^2\left(\sqrt{c}\,\frac{(x - ct - x_0)}{2}\right). \tag{4.7}$$

图5 总结了 KdV方程的数值结果。面板 (a) 展示了数值解（ML-TT 方法）与解析单孤子

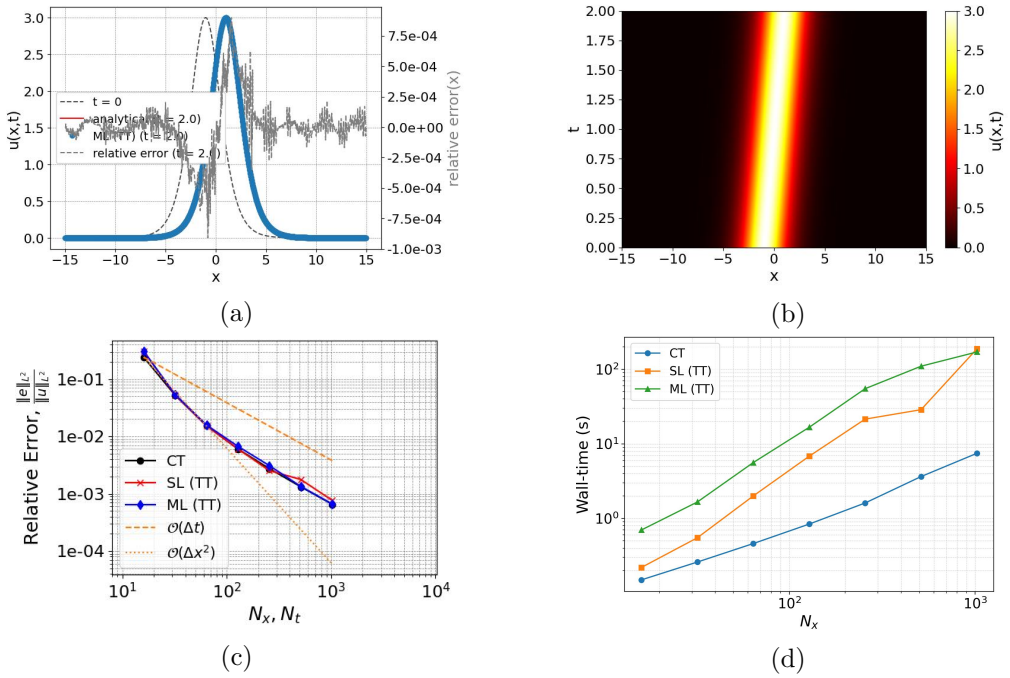


图 5：单孤子情形下 KdV 方程（4.6）的数值解。离散化采用 $N_x = N_t = 2^{10}$，时间步长为 $\Delta t = 2/N_t$。（a）在 $t = 2$处数值解与解析解的比较；（b）数值解的时空等高线图；（c）网格细化时数值误差的收敛情况：在粗网格上可观察到二阶空间精度 $\mathcal{O}(\Delta x^2)$，而在较细网格上，误差主要由一阶时间误差贡献决定$\mathcal{O}(\Delta t)$；（d）经典时间步进方法（CT）、单层（SL）和多层（ML）时空方法的实际运行时间。

解在 $t = 2$ 处的高度一致性，该结果采用均匀的时空离散，并使用 $N_x = N_t = 2^{10}$ 和时间步长 $\Delta t = 2/N_t$ 获得。数值剖面与解析剖面几乎无法区分，证实了所提方法的准确性。

面板（b）展示了（x，t）平面上的数值解，体现了平滑的时空演化以及孤子的稳定传播，未见明显的数值伪影。

图 (c) 显示了在 $t = 2$处评估的数值误差的收敛行为。结果表明，在网格细化过程中，误差按 $\mathcal{O}(\Delta t,\ \Delta x^2)$ 收敛。这一行为与时间离散采用隐式欧拉格式（其精度为一阶）以及空间离散采用二阶中心有限差分格式相一致。

表 5 中的比较分析证实了张量列车时空方法在 KdV 测试算例中的两项主要优势。首先，单层和多层求解器的解精度与经典时间步进参考解相当（几乎完全一致），表明舍入误差得到了控制。其次，多层求解器以少于单层求解器的牛顿迭代次数达到了这一精度，凸显了多层初始化的优势。

对于混双曲和色散问题，例如 KdV方程，求解整体式

$N_x \times N_t$	牛顿迭代		运行时间（秒）			最大秩 χ		相对误差 $\|e\|_{L^2}/\|u\|_{L^2}$			$\|e\|_{L^2}$
	SL	ML	SL	ML	CT	SL	ML	SL	ML	CT	CT
$2^4 \times 2^4$	3	4	0.22	0.70	0.15	8	7	3.07e-01	3.07e-01	2.42e-01	1.18e+00
$2^5 \times 2^5$	3	3	0.55	1.66	0.26	10	10	5.29e-02	5.42e-02	5.23e-02	2.56e-01
$2^6 \times 2^6$	3	3	2.00	5.61	0.46	13	13	1.53e-02	1.58e-02	1.53e-02	7.50e-02
$2^7 \times 2^7$	3	2	6.82	16.66	0.84	13	13	6.05e-03	6.71e-03	6.03e-03	2.95e-02
$2^8 \times 2^8$	5	3	21.44	54.87	1.61	13	13	2.53e-03	3.09e-03	2.74e-03	1.34e-02
$2^9 \times 2^9$	5	3	28.70	109.83	3.64	13	13	1.77e-03	1.34e-03	1.32e-03	6.49e-03
$2^{10} \times 2^{10}$	13	3	188.14	170.50	7.47	13	13	7.65e-04	6.62e-04	6.53e-04	3.20e-03

表5：KdV 单孤子问题的数值结果。张量列车求解器参数为$\varepsilon_{\mathrm{TT}} = 10^{-6}$、 $\varepsilon_{\mathrm{DMRG}} = 10^{-3}$和 $\varepsilon_{\mathrm{Newton}} = 10^{-3}$。DMRG扫描的最大次数固定为 3，牛顿迭代最多进行 20 次，线搜索参数为 $s = 0.8$。多层方法采用 $n_l = \min(q_x, q_t) - 2$ 个层级。这里，SL 表示单层张量列车方法，ML 表示多层张量列车方法。误差使用 $\|u\|_{L^2} = 11.2$进行归一化。计算域长度为 $L = 30$ ，孤子速度为 $c = 1$。蒂科诺夫正则化参数设为 $\alpha = 10^{-12}$。

随着时空分辨率提高，TT 格式下的时空系统变得越来越难以求解。具体而言，基于 DMRG 的牛顿校正求解过程中产生的局部线性系统通常严重病态，且随着时空网格细化，其条件性迅速恶化。这种病态性会导致牛顿迭代停滞或失败，因此必须采用适当的稳定化措施。

为缓解这一困难，在局部 DMRG 求解层面应用蒂霍诺夫正则化，以稳定投影线性系统，同时保留整体低秩结构。尽管这种正则化提高了单层和多层公式的稳健性，但仅靠它仍不足以保证在细网格上的收敛。

通过多层策略实现了稳健的收敛：在该策略中，粗网格时空解用于初始化逐步细化网格上的牛顿迭代。这种分辨率上的非线性延拓显著减轻了病态性的影响，并实现了近似与网格无关的收敛。相比之下，单层时空 TT 求解器表现出迅速恶化的收敛行为，并且在离散化足够精细时无法收敛。

从数值结果来看，多层公式在所有测试分辨率下所需的牛顿迭代次数大致相同（通常为三次）。相比之下，单层方法中的牛顿迭代次数随着网格细化而持续增加。对于规模更大的问题（例如，自由度超过 $2^{10} \times 2^{10}$ 的时空网格），该方法无法收敛。

对于所有 KdV 实验，TT 键维数固定为 13。尽管可以采用自适应秩策略，但我们的结果表明，对于此类问题，增加键维数并不能提高精度。相反，限制键维数能够同时提升收敛稳健性和计算效率。因此，所选键维数在精度与稳定性之间实现了有效平衡。

只要张量列车秩保持有界，基于 TT 的方法的计算复杂度就能得到控制。具体而言，时空细化会使经典求解器的计算成本迅速增长（通常呈指数级），而基于 TT 的方法则

在牛顿迭代次数近乎恒定的前提下，其计算开销相对于时空自由度总数呈对数—线性增长。我们的实验始终观察到这一现象：将问题规模从 9 个量子比特增至 10 个量子比特时，运行时间仅略有增加；相比之下，单层公式在更早的细化阶段就已显示出计算成本近乎翻倍的趋势。

值得一提的是，尽管本研究仅限于一维空间问题，但预计多层张量列车时空框架的优势在更高维度下会更加显著。经典方法的复杂度通常会随空间维度呈指数增长，而只要张量秩保持适中，基于 TT 的方法便能够缓解这一问题。因此，将方法拓展至二维和三维问题是未来研究的自然方向。

5. 结论。本文从务实的角度重新审视了用于非线性偏微分方程的整体式时空求解器。尽管张量列车表示能够大幅压缩全局时空离散，但我们的结果表明，仅靠压缩并不能解决核心难题：实现稳健的非线性收敛。在对流占优和双曲型区域中，即使采用低秩表示，单层整体牛顿迭代也经常停滞。

本文的核心贡献是提出一种直接嵌入张量列车框架的非线性多层策略，其唯一目的是为整体牛顿求解构造可靠的初始猜测。每一层均独立求解直至收敛，然后将结果延拓至下一层细化网格。该方法不进行残差校正或多层循环。这一区别至关重要：该方法的设计目标不仅是加速线性收敛，还要防止非线性求解失败。另一个关键组成部分是对牛顿–DMRG迭代进行正则化。对于双曲型问题，张量列车求解器中产生的投影局部系统往往近乎奇异。若不进行正则化，牛顿更新会变得不稳定并放大噪声。采用适度的蒂霍诺夫稳定化后，既能恢复收敛，又能保留低秩结构。在这种情况下，正则化并非可有可无。本文所呈现的全部数值实验均限于一维空间，此时经典时间步进已经表现良好。即使在这一有利的维度设置下，多层张量列车求解器仍表现出更强的稳健性和有竞争力的扩展性。更重要的是，其根本动机在于维度：经典方法的复杂度随空间维度呈指数增长，而基于 TT 的表述在秩增长适中的情况下仍能实现指数级压缩。本文提出的多层策略与维度无关，预计其在更高维度问题中将发挥越来越重要的作用。

整体式时空求解器在求解非线性偏微分方程时的有效性，并不单单取决于压缩，还取决于如何在张量列车框架中稳定并初始化牛顿法。将非线性多层策略与正则化牛顿–DMRG迭代相结合，为解决问题提供了可行的途径，尤其适用于具有挑战性的双曲动力学问题。

致谢。 最后三位作者衷心感谢编号为 KUEXT-8434000491 的资助。该资助由哈利法大学、技术创新研究所和阿联酋核能公司合作提供。本研究由阿联酋核技术中心开展，该中心由哈利法科技大学、阿联酋核能公司和联邦核监管局共同合作建立。

附录 A. 算子的张量列车表示。设 $\mathcal{T}^{(n)}(l,d,u;\,a_1,a_2)$ 是一个 $2^n\times 2^n$ 托普利兹矩阵，其次对角线为 l、对角线为 d、超对角线为 u，并由边界条件 a_1,a_2（方程 (3.5)）修正。对于周期边界，将 $\mathcal{T}^{(n)}(l,d,u;\,a_1,a_2)$ 替换为

大小为 $2^n \times 2^n$ 的循环三对角矩阵 $\mathcal{T}_{(n)c}(l,d,u)$，定义如下：

$$\mathcal{T}^{(n)}(l,d,u;a_1,a_2) = \underbrace{\begin{pmatrix} a_1l+d & u & & & \\ l & d & u & & \\ & \ddots & \ddots & \ddots & \\ & & l & d & u \\ & & & l & a_2u+d \end{pmatrix}}_{2^n\times 2^n},\ \mathcal{T}_c^{(n)}(l,d,u) = \underbrace{\begin{pmatrix} d & u & & & l \\ l & d & u & & \\ & \ddots & \ddots & \ddots & \\ & & l & d & u \\ u & & & l & d \end{pmatrix}}_{2^n\times 2^n}.$$

按照 [14], 张量列车表示 $\mathcal{T}_{(n)}(l,\ d,\ u;\ a_1,\ a_2)$和 $\mathcal{T}_{(n)c}(l,\ d,\ u)$的步骤如下：

$$\mathcal{T}^{(n)}(l,d,u;a_1,a_2) = \begin{bmatrix} I_1 & I & J^{'} & J & I_2 \end{bmatrix} \bowtie \begin{bmatrix} I_1 & & & & \\ & I & J^{'} & J & \\ & & J & & \\ & & & J^{'} & \\ & & & & I_2 \end{bmatrix}^{\bowtie(n-2)} \bowtie \begin{bmatrix} a_1I_1 \\ dI+uJ+lJ^{'} \\ lJ \\ uJ^{'} \\ a_2I_2 \end{bmatrix},$$

$$\mathcal{T}_c^{(n)}(l,d,u) = \begin{bmatrix} I & J^{'}+J & J+J^{'} \end{bmatrix} \bowtie \begin{bmatrix} I & J^{'} & J \\ & J & \\ & & J^{'} \end{bmatrix}^{\bowtie(n-2)} \bowtie \begin{bmatrix} dI+uJ+lJ^{'} \\ lJ \\ uJ^{'} \end{bmatrix}.$$

其中,

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, J = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, J^{'} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, I_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, I_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

我们用 $\bowtie$表示两个张量之间的秩核乘积，其定义为矩阵乘法与逐元素克罗内克积 [14]的组合。五对角矩阵 $\mathcal{T}_{(n)\,5}(l_2,\ l_1,\ d,\ u_1,\ u_2;\ a_{1,\ 1},\ a_{1,\ 2},\ a_{1,\ 3},\ a_{2,\ 1},\ a_{2,\ 2},\ a_{2,\ 3})$，以及循环五对角矩阵

$\mathcal{T}_{(n)\ c5}(l_2,\ l_1,\ d,\ u_1,\ u_2)$ 的大小为 $2^n \times 2^n$：

$$\mathcal{T}_5^{(n)}(l_2,l_1,d,u_1,u_2;a_{1,1},a_{1,2},a_{1,3},a_{2,1},a_{2,2},a_{2,3}) = \begin{bmatrix} a_{1,1}+d & a_{1,2}+u_1 & u_2 & & & & \\ a_{1,3}+l_1 & d & u_1 & u_2 & & & \\ l_2 & l_1 & d & u_1 & u_2 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & l_2 & l_1 & d & u_1 & u_2 \\ & & & l_2 & l_1 & d & a_{2,3}+u_1 \\ & & & & l_2 & a_{2,2}+l_1 & a_{2,1}+d \end{bmatrix}, \tag{A.1}$$

$$\mathcal{T}_{c5}^{(n)} = \begin{bmatrix} d & u_1 & u_2 & & & l_2 & l_1 \\ l_1 & d & u_1 & u_2 & & & l_2 \\ l_2 & l_1 & d & u_1 & u_2 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & l_2 & l_1 & d & u_1 & u_2 \\ u_2 & & & l_2 & l_1 & d & u_1 \\ u_1 & u_2 & & & l_2 & l_1 & d \end{bmatrix},$$

$$\mathcal{T}_5^{(n)} = \begin{bmatrix} I_1 & I & J^{'} & J & I_2 \end{bmatrix} \bowtie \begin{bmatrix} I_1 & & & & \\ & I & J^{'} & J & \\ & & J & & \\ & & & J^{'} & \\ & & & & I_2 \end{bmatrix}^{\bowtie(n-2)} \bowtie \begin{bmatrix} a_{1,1}I_1 + a_{1,2}J + a_{1,3}J^{'} \\ l_1J^{'} + dI + u_1J \\ l_1J + l_2I \\ u_1J^{'} + u_2I \\ a_{2,1}I_2 + a_{2,2}J^{'} + a_{2,3}J \end{bmatrix}, \tag{A.2}$$

$$\mathcal{T}_{c5}^{(n)} = \begin{bmatrix} I & J^{'}+J & J+J^{'} \end{bmatrix} \bowtie \begin{bmatrix} I & J^{'} & J \\ & J & \\ & & J^{'} \end{bmatrix}^{\bowtie(n-2)} \bowtie \begin{bmatrix} l_1J^{'} + dI + u_1J \\ l_1J + l_2I \\ u_1J^{'} + u_2I \end{bmatrix}, \tag{A.3}$$

算子 D_t、J_t、K_t、D_x、D_{xx}、D_{xxx} 的矩阵表示如下：

$$D_t = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}, \quad J_t = \frac{1}{2}\begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ & \ddots & \ddots & \\ & & 1 & 1 \end{bmatrix}, \quad K_t = \frac{1}{4}\begin{bmatrix} 1 & & & & \\ 2 & 1 & & & \\ 1 & 2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 2 & 1 \end{bmatrix},$$

$$D_x = \frac{1}{2\Delta x}\begin{bmatrix} -a_1 & 1 & & & \\ -1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & a_2 \end{bmatrix}, \quad D_{xx} = \frac{1}{\Delta x^2}\begin{bmatrix} a_1 - 2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & a_2 - 2 \end{bmatrix},$$

$$D_{tt} = \begin{bmatrix} 1 & & & & \\ -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix}, \quad D_{xxx} = \frac{d^3u}{dx^3} \approx \frac{1}{2\Delta x^3}\begin{bmatrix} 0 & -2 & 1 & & & & \\ 2 & 0 & -2 & 1 & & & \\ -1 & 2 & 0 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & -1 & 2 & 0 & -2 & 1 \\ & & & -1 & 2 & 0 & -2 \\ & & & & -1 & 2 & 0 \end{bmatrix},$$

其中 a_1 和 a_2 分别表示左右边界处的边界条件类型，定义见方程（3.5）。算子 D_t、J_t、K_t、D_{tt}、D_x、D_{xx}、D_{xxx} 的张量列车表示为：

$$D_t = \mathcal{T}^{(q_t)}(-1,1,0;0,0), \quad J_t = \frac{1}{2}\mathcal{T}^{(q_t)}(1,1,0;0,0), \quad K_t = \frac{1}{4}\mathcal{T}_5^{(q_t)}(1,2,1,0,0),$$
$$D_{tt} = \mathcal{T}_5^{(q_t)}(1,-2,1,0,0), \quad D_x = \frac{1}{2\Delta x}\mathcal{T}^{(q_x)}(-1,0,1;a_1,a_2),$$
$$D_{xx} = \frac{1}{\Delta x^2}\mathcal{T}^{(q_x)}(1,-2,1;a_1,a_2), \quad D_{xxx} = \frac{1}{2\Delta x^3}\mathcal{T}_{c5}^{(q_x)}(-1,2,0,-2,1).$$

参考文献

[1] M. J. Ablowitz and A. Zeppetella, *Explicit solutions of fisher's equation for a special wave speed*, 《数学生物学公报》, 41（1979年）, 第835–840页, https://www.sciencedirect.com/science/article/pii/ S0092824079800208。[2] D. Adak, M. E. Danis, D. P. Truong, K. Ø. Rasmussen, and B. S. Alexandrov, *Tensor network space-time spectral collocation method for solving the nonlinear convection diffusion equation*, 《科学计算杂志》, 103（2025年）, 第46页, https://doi.org/10.1007/s10915-025-02860-x。[3] D. Adak, D. P. Truong, G. Manzini, K. Ø. Rasmussen, and B. S. Alexandrov, *Tensor network space-time spectral collocation method for time dependent convection-diffusion-reaction equations*, 2024年, https: // arxiv.org/abs/2402.18073。[4] L. Arenstein, M. Mikkelsen, and M. Kastoryano, *Fast and flexible quantum-inspired differential equation solvers with data integration* 2025 2505 17046 , , https://arxiv.org/abs/ 。。[5] A. Bratsos, *A numerical method for the one-dimensional sine-Gordon equation*, 偏微分方程数值方法, 24（2008年）, 第833–844页, https://doi.org/10.1002/num.20292。[6] T. Breiten, S. Dolgov, and M. Stoll, *Solving differential Riccati equations: A nonlinear space-time method using tensor trains* 11 2021 407 429, , 数值代数、控制与优化, （）, 第–页 https://doi. org/10.3934/naco.2020034。[7] S. V. Dolgov, B. N. Khoromskij, and I. V. Oseledets, *Fast solution of parabolic problems in the tensor train/quantized tensor train format with initial application to the fokker–planck equation* , SIAM 科学计算杂志, 34（2012年）, 第A3016–A3038页, https://doi.org/10.1137/120864210。

[8] S. V. Dolgov and D. V. Savostyanov, *Alternating minimal energy methods for linear systems in higher dimensions* 36 2014 2248 2271 , SIAM 科学计算杂志, （）, 第A–A页。[9] M. Dwivedi and T. Sarkar, *Convergence of a conservative crank–nicolson finite difference scheme for the kdv equation* 2025 10 1002 70035 （）, https://doi.org/ . /mma. 。即将出版/提前在线发表。[10] R. A. Fisher, *The wave of advance of advantageous genes*, 《优生学年鉴》, 7（1937年）, 第355–369页, https: //doi.org/10.1111/j.1469-1809.1937.tb02153.x。[11] H. Holden, U. Koley, and N. H. Risebro, *Convergence of a fully discrete finite difference scheme for the Korteweg–de Vries equation* 35 2014 1047 1077, , IMA 数值分析杂志, （）, 第–页 https: //doi.org/10.1093/imanum/dru040。[12] S. Holtz, T. Rohwedder, and R. Schneider, *The alternating linear scheme for tensor optimization in the tensor train format* 34 2012 683 713 , SIAM 科学计算杂志, （）, 第A–A页。[13] G. Horton and S. Vandewalle, *A space-time multigrid method for parabolic partial differential equations*, SIAM 科学计算杂志, 16（1995年）, 第848–864页, https://doi.org/10.1137/0916050。[14] V. A. Kazeev and B. N. Khoromskij, *Low-rank explicit QTT representation of the Laplace operator and its inverse* 33 2012 742 758 , SIAM 矩阵分析与应用杂志, （）, 第–页。[15] B. N. Khoromskij and C. Schwab, *Tensor-structured galerkin approximation of parametric and stochastic elliptic PDEs* 33 2011 364 385 , SIAM 科学计算杂志, （）, 第–页。[16] M. Mohr and R. Wienands, *Cell-centred multigrid revisited*, 《科学计算与可视化》, 7（2004年）, 第129–140页, https://doi.org/10.1007/s00791-004-0137-0。[17] N. M. Newmark, *A method of computation for structural dynamics*, 《工程力学杂志》, 85（1959年）, 第67–94页, https://doi.org/10.1061/JMCEA3.0000098, https://ascelibrary.org/doi/10.1061/ JMCEA3.0000098。[18] I. Oseledets and E. Tyrtyshnikov, *TT-cross approximation for multidimensional arrays*, 《线性代数及其应用》, 432（2010年）, 第70–88页, https://doi.org/10.1016/j.laa.2009.07.024。[19] I. Oseledets and E. Tyrtyshnikov, *TT-cross approximation for multidimensional arrays*, 《线性代数及其应用》, 432（2010年）, 第70–88页。[20] I. V. Oseledets, *Tensor-train decomposition*, SIAM 科学计算杂志, 33（2011年）, 第2295–2317页。[21] I. V. Oseledets and S. V. Dolgov, *Solution of linear systems and matrix inversion in the TT-format*, SIAM 科学计算杂志, 34（2012年）, 第A2718–A2739页, https://doi.org/10.1137/110833142。[22] R. D. Peddinti, S. Pisoni, N. Rapaka, M. K. Riahi, E. Tiunov, and L. Aolita, *Quantum-inspired space-time pde solver and dynamic mode decomposition* 2510 21767, 2025 , arXiv 预印本 arXiv: 。（）。[23] N. R. Rapaka and R. Samtaney, *An efficient poisson solver for complex embedded boundary domains using the multi-grid and fast multipole methods* 410 2020 109387, , 《计算物理杂志》, （）, 第页 https: //doi.org/10.1016/j.jcp.2020.109387。[24] M. K. Riahi, *Pitsbicg: Parallel in time stable bi-conjugate gradient algorithm*, 《应用数值数学》, 181（2022年）, 第225–233页, https://doi.org/10.1016/j.apnum.2022.06.004。[25] A. Rodgers and D. Venturi, *Implicit integration of nonlinear evolution equations on tensor manifolds*, 《科学计算杂志》, 97（2023年）, 第33页, https://doi.org/10.1007/s10915-023-02352-w, https: //doi.org/10.1007/s10915-023-02352-w。[26] U. Schollwöck, *The density-matrix renormalization group in the age of matrix product states*, 《物理年鉴》, 326（2011年）, 第96–192页, https://doi.org/10.1016/j.aop.2010.09.012。2011年1月特刊。[27] P. Wesseling, *An Introduction to Multigrid Methods*, J. Wiley, 1992年。[28] W. L. Wood, *An exact solution for Burger's equation*, 《工程数值方法通讯》, 22（2006年）, 第797–798页, https://doi.org/10.1002/cnm.850, https://doi.org/10.1002/cnm.850。[29] N. J. Zabusky and M. D. Kruskal, *Interaction of "solitons" in a collisionless plasma and the recurrence of initial states* 15 1965 240 243, 10 1103 15 240 , 《物理评论快报》, （）, 第–页 https://doi.org/ . /PhysRevLett. . 。