arXiv:2304.05592v1 [cs.MS] 12 Apr 2023

结合可微编程和机器学习的学习型多物理场反演

Mathias Louboutin[1, *], Ziyi Yin[1, *], Rafael Orozco[1], Thomas J. GradyII[1], Ali Siahkoohi[2], Gabrio Rizzuti[3], Philipp A. Witte[4], Olav Møyner[5], Gerard J. Gorman[6], Felix J. Herrmann[1]

[1] 佐治亚理工学院, [2] 莱斯大学, [3] 乌得勒支大学医学中心, [4] 微软, [5] SINTEF Digital, [6] 伦敦帝国理工学院

摘要

我们介绍用于计算《地球物理学》以及更广泛的涉及波动方程的反问题（例如地震和医学超声）、采用学习先验的正则化，以及用于多相流模拟的神经代理模型的地震成像与建模/监测实验室（SLIM）开源软件框架。通过集成多个抽象层，我们的软件旨在兼具可读性和可扩展性。这使研究人员能够以抽象的方式轻松构建问题，同时利用高性能计算领域的最新进展。我们通过构建一个可扩展的原型来展示和演示我们的设计原则及其优势：该原型利用时移交叉井地震数据进行渗透率反演；除了耦合波动物理和多相流之外，还涉及机器学习。

动机

得益于高性能计算（HPC）技术的重大进步，计算（勘探）地球物理学在过去几十年里取得了巨大飞跃。例如，这些进展促成了基于波动方程的反演技术的应用，例如全波形反演（FWI）和逆时偏移（RTM）。由于这些技术遵循波动物理，因此在复杂地质条件下实现了更优异的成像。尽管这些技术无疑位居最先进的成像技术之列，但除少数例外情况外——最值得一提的是 iWave++ (Sun 和 Symes，2010年)、Julia Devito 反演框架（JUDI.jl 地震成像与建模实验室（SLIM）的 P. A. Witte、Louboutin、Kukreja 等人（2019年）；Mathias Louboutin 等人（2023年）），以及 Chevron 的COFII (Washbourne 等人，2021年)——其实现依赖于整体式低级（C/Fortran）实现。因此，由于缺乏抽象和现代编程结构，这些低级实现难以维护且维护成本极高，尤其是在性能考量优先于最佳软件实践时。尽管这些实现设计选择能针对特定问题（例如全波形反演）生成高性能代码，却往往阻碍新算法的实现，例如基于不同目标函数或约束条件的算法，也妨碍现有代码库与外部软件库的耦合。例如，将基于波动方程的反演与机器学习框架结合，或将波动物理与多相流体流动求解器耦合，都被认为具有挑战性且成本高昂。因此，我们的行业面临失去创新能力的风险，而能源转型带来的挑战更使这一情况雪上加霜。

*这些作者贡献相同。通讯作者：mlouboutin3@gatech.edu

设计原则

为了解决当前软件实现中的这些重大缺陷及其对发展的阻碍，我们着手开发一个高性能软件框架。例如，我们使用Devito实现的波传播算子（M. Louboutin 等人，2019年；Luporini 等人，2020年）已被承包商和石油和天然气巨头用于生产，同时支持针对多物理场和机器学习问题的算法进行快速、低成本、可扩展且可互操作的开发，并可在多种不同芯片组（例如 ARM、英特尔、POWER）和图形加速器（例如英伟达）上运行。为此，我们采用当代软件设计实践，包括高层抽象、软件设计原则，以及使用 Python（Rossum 和 Drake，2009年）和 Julia （Bezanson 等人，2017年）等现代编程语言。我们还广泛使用领域特定语言（DSL）提供的抽象，例如 Rice 向量库（RVL，Padula、Scott 和 Symes，2009年）以及统一形式语言（UFL，Rathgeber 等人，2016年；Alnaes 等人，2015年），并采用由开创性的开源项目 Madagascar（Fomel 等人，2013年）引入的可复现研究实践。该项目成功运用了版本控制，以及基于软件构建工具 `SCons`的抽象。

为应对性能至关重要的环境中现代软件设计所面临的挑战，除关注点分离这一基本原则外，我们还遵循三项关键原则。首先，我们采用数学语言来指导抽象。数学简洁、明确、易于理解，并能为

- **波动物理**，通过偏微分方程来实现，正如 Devito 所采用的方式。Devito 借助符号 Python（SymPy）（Meurer 等，2017）来定义偏微分方程。给定符号表达式后，Devito 会自动生成高度优化、可能采用域分解的并行 C 代码，以近乎最优的性能适配可用硬件，用于三维声学、倾斜横向各向同性或弹性波方程；
- **线性代数**，通过无矩阵线性算子实现，例如JUDI.jl （P. A. Witte、Louboutin、Kukreja 等，2019；Mathias Louboutin 等，2023）——一种用于基于波动方程的建模与反演的高级线性代数领域专用语言。这些理念可追溯至 SPOT（Berg 和 Friedlander，2009），较新的实现包括JOLI.jl （Modzelewski 等，2023），以 Julia 编写；以及 PyLops ，以 Python 编写（Ravasi 和 Vasconcelos，2020）。
- **优化**，通过定义需要最小化的目标函数（也称为损失函数），并借助SlimOptim.jl （MathiasLouboutin、Yin 和Herrmann2022b）进行求解，同时满足数学约束，这些约束可通过SetIntersectionProjection.jl （Peters 和 Herrmann 2019；Peters、Louboutin 和 Modzelewski 2022）施加。

其次，我们利用基于波动方程的反演问题中的层次结构，这自然会带来关注点分离。在最高层次上，我们处理线性算子，具体而言，是基于波动的反演中的矩阵自由雅可比矩阵；我们使用 JUDI.jl 和 SegyIO.jl进行并行文件输入/输出，可在本地部署（Lensink 等，2023），也可通过 JUDI4Cloud.jl 在云端（Azure）使用（Mathias Louboutin、Yin 和 Herrmann，2022a）。在中低层次上，我们广泛使用 Devito （M. Louboutin 等，2019；Luporini 等，2020）——一种用于基于模板的时域有限差分计算的即时编译器，多年来 SLIM 一直参与其开发。

第三，我们以 Mike Innes 等人（2019 年倡导的可微编程原则，以及 D. Li 等人（2020 年提出的侵入式自动微分为基础，将波动物理学与机器学习框架和多相流相结合。具体而言，我们通过使用链式法则来采用自动微分（自动微分），其中包括允许用户添加导数规则的抽象，例如 ChainRules.jl（White 等人，2022年、2023 年）。

在2015年首届全波形反演研讨会上，我们在 SLIM 开始阐述这些设计原则（Lin 和 Herrmann，2015），经过多年的发展，这些原则最终形成了面向时间谐波全波形反演的可扩展并行软件框架（Silva 和 Herrmann，2019）、面向时域逆时偏移和全波形反演的软件框架（P. A. Witte、Louboutin、Kukreja 等，2019；Mathias Louboutin 等，2023），以及面向抽象化全波形反演的软件框架（Mathias

Louboutin 等，2022），从而实现了与机器学习的连接。除了开发基于波动方程的反演软件之外，近年来我们团队还参与了可扩展机器学习解决方案的开发，包括 Julia 软件包 InvertibleNetworks.jl （P. Witte 等，2023、2023），该软件包实现了具有内存效率的可逆深度神经网络，例如（条件）归一化流（NFs，Rezende 和 Mohamed，2015），以及可扩展的分布式傅里叶神经算子（FNOs，Z. Li 等，2020），后者由 dfno 软件包实现（Grady 等，2022；Grady、Infinoid 和 Louboutin，2022）。下面将对所有这些内容进行更详细的介绍。

为了说明这些设计原则如何促成复杂学习型耦合反演问题的解决方案，我们将在接下来的章节中讨论时移地震数据的空间渗透率分布端到端反演（D. Li 等，2020）。如图 1所示，该反演问题相当复杂，其解决方案可以说得益于上述三项设计原则。在这一表述中，渗透率的潜在表示通过一系列非线性运算，最终映射到时移地震数据。在本文接下来的部分，我们将详细介绍如何实现这一学习型反演问题中的不同组件，从而开展耦合反演。所呈现的结果是初步的，代表了依据这些设计原则开展研究的一个阶段性成果。

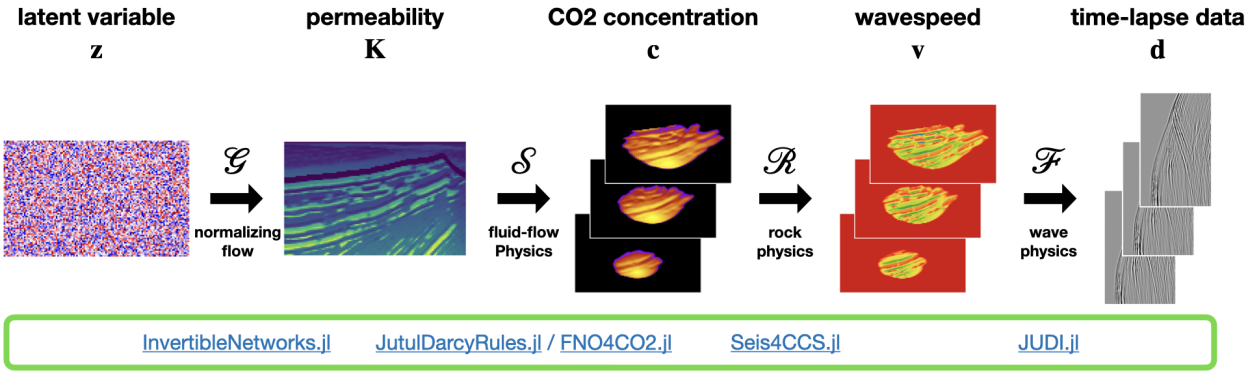


图 1：多物理场正演模型。渗透率$\mathbf{K}$ 由高斯噪声通过预训练归一化流（pretrained NF）$\mathcal{G}$ 生成，随后经过通过 $\mathcal{S}$ 进行的两相流模拟、由 $\mathcal{R}$ 表示的岩石物理，以及通过波动物理进行的时移地震数据模拟 $\mathcal{F}$。

学习型时移端到端渗透率反演

应对气候变化和推动能源转型，需要解决日益复杂的问题。构建用于地质 CO_2 和/或 H_2 储存的地震监测系统便属于这类问题。为了说明数学启发的抽象如何发挥作用，我们研究如何利用井间时移数据反演渗透率（实验设置见图 2 ），其中涉及：(i) 使用 Jutul.jl （Møyner 等，2023）将波动物理与两相（盐水/CO_2）流动耦合。Jutul.jl 是基于 Julia 的先进油藏建模软件；(ii) 使用 InvertibleNetworks.jl，通过归一化流实现学习型正则化；(iii) 使用傅里叶神经算子，为流体流动模拟构建学习型代理模型。这类反演问题尤其具有挑战性，因为它需要结合不同类型的物理过程，从咸水含水层的井间数据中估计 CO_2 羽流过去、当前和未来的饱和度与压力分布。在后续章节中，我们将展示如何使用图 1所列的独立软件包反演时移数据。

基于波动方程的反演

凭借其对 CO_2 羽流无与伦比的分辨能力，主动源时移地震可以说是监测地质封存的首选成像方式（Ringrose 2020）。对于一次

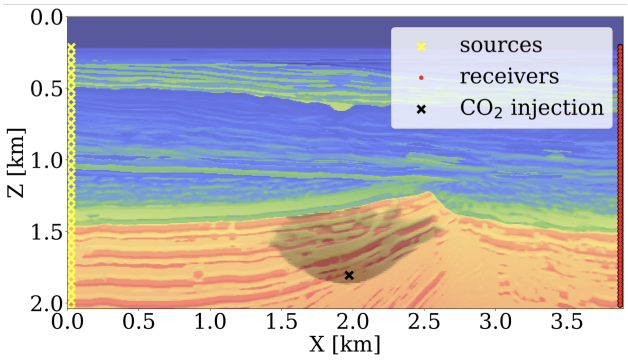


图 2：实验设置。模型中央的黑色 × 符号表示 CO_2 注入位置。地震源位于模型左侧（以黄色 × 符号表示），接收器位于模型右侧（以红点表示）。我们以灰色叠加显示压缩波速，以及模拟了 18年后的 CO_2 饱和度。

时移采集期次，最简单情况下的全波形反演（FWI）涉及最小化观测数据与合成数据之间的 ℓ_2-范数失配/损失函数，即我们有

$$\underset{\mathbf{m}}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{F}(\mathbf{m})\mathbf{q} - \mathbf{d}\|_2^2 \quad \text{where} \quad \mathbf{F}(\mathbf{m}) = \mathbf{P}_r\mathbf{A}(\mathbf{m})^{-1}\mathbf{P}_s^\top. \tag{1}$$

在此表述中，符号$\mathbf{F}(\mathbf{m})$ 表示由慢度平方参数化的正演建模算子（波动物理），$\mathbf{m}$。该正演算子作用于震源，由源注入算子$\mathbf{P}_s^\top$ 与 $^\top$ 组成；后者表示转置算子，通过$\mathbf{A}(\mathbf{m})^{-1}$求解离散化波动方程，并通过线性算子$\mathbf{P}_r$限制到接收器位置。向量$\mathbf{q}$ 表示地震源，向量$\mathbf{d}$ 包含在接收器位置采集的单个采集期次地震数据。由于我们遵循数学表达，对未知慢度平方$\mathbf{m}$ 进行反演的相应 Julia 代码（使用JUDI.jl ）如下

```
# Forward modeling to generate seismic data.
Pr = judiProjection(recGeometry)  # setup receiver
Ps = judiProjection(srcGeometry)  # setup sources
Ainv = judiModeling(model)        # setup wave-equation solver
F = Pr * Ainv * Ps'               # forward modeling operator
d = F(m_true) * q                 # generate observed data
# Gradient descent to invert for the unknown squared slowness.
for it = 1:maxiter
    d0 = F(m) * q                 # generate synthetic data
    J = judiJacobian(F(m), q)     # setup the Jacobian operator of F
    g = J' * (d0 - d)             # gradient w.r.t. squared slowness
    m = m - t * g                 # gradient descent with steplength t
end
```

为了得到这种简洁且抽象的全波形反演表述，我们在Devito 中对传播算子采用了层次化抽象，并在JUDI.jl中使用了线性代数工具，包括针对 F 及其雅可比矩阵 J的矩阵无关实现。虽然上述独立实现可用于稀疏性促进的地震反演（P. A. Witte、Louboutin、Luporini 等，2019；Herrmann、Siahkoohi 和 Rizzuti，2019；Yang 等，2020；Rizzuti 等，2020、2021；Siahkoohi、Rizzuti 和 Herrmann，2020a、2020b、2020c；Yin、Louboutin 和 Herrmann，2021；Yin 等，2023）

以及医学反演（Yin 等（2020）；Orozco 等（2021，2023）；Orozco、Louboutin 等（2023）），该实现依赖于手工推导的雅可比矩阵伴随算子 `J'` 和损失函数导数。尽管这种方法可行，但如果只依赖手工推导的导数，那么在使用机器学习模型或需要将波动方程与多相流方程耦合时，过程可能会变得繁琐。

为适应这种情况，我们使用 Julia 的可微编程生态系统，其中包含使用自动微分的工具，并可通过 ChainRules.jl 添加微分规则。借助这一工具，可以教会自动微分系统如何对JUDI.jl求导，具体通过以下针对正向传播算子的微分规则实现

```
# Custom AD rule for wave modeling operator.
function rrule(::typeof(*), F::judiModeling, q)
y = F * q                    # forward modeling
# The pullback function for gradient calculations.
pullback(dy) = NoTangent(),  judiJacobian(F, q)' * dy,  F' * dy
return y,  pullbackend
```

在此规则中， `pullback` 函数以数据残差 `dy` 为输入，并输出相对于算子 `*` （无梯度）、模型参数和源分布的梯度。借助此微分规则，上述梯度下降算法可实现如下：

```
# Define the loss function.
loss(m) = .5f0 * norm(F(m) * q - d)^2f0
# Gradient descent to invert for the squared slowness.
for it = 1:maxiter
    g = gradient(loss, m)[1]      # gradient computation via AD
    m = m - t * g                 # gradient descent with steplength t
end
```

与原始实现相比，此代码只需要 `F(m)` 以及函数 `loss(m)`。借助上述 `rrule`，Julia 的自动微分系统[1] 能够计算梯度（第 5行）。除了保持高性能之外，即我们仍然利用伴随状态法计算梯度，这种方法的优势在于它提供了更大的灵活性，例如，在慢度平方通过预训练神经网络或多相流模拟的输出进行参数化的情况下。在下一节中，我们将展示训练好的归一化流如何作为先验来提高全波形反演的质量。

深度先验与归一化流

归一化流（NFs）是一类生成模型，利用可逆深度神经网络架构从训练样本中学习复杂分布（Dinh、Sohl-Dickstein 和 Bengio（2016））。例如，在地震反演应用中，我们希望近似地球模型的分布，并将其用作下游任务的先验。归一化流通过一系列可训练的非线性可逆层，将目标分布（即地球模型）中的样本映射为零均值、单位标准差的高斯噪声。训练完成后，可以重新采样新的高斯噪声，并将其通过这组层的逆序列，以获得来自目标分布的新生成样本。归一化流是地震应用中颇具吸引力的生成模型选择（Zhang 和 Curtis（2020，2021）；Zhao、Curtis 和 Zhang（2021）；Siahkoohi 和 Herrmann（2021）；Siahkoohi 等（2021，2023）；Siahkoohi、Rizzuti 和 Herrmann（2022）），因为它们采样速度快，而且由于其固有的可逆性，无需在反向传播过程中

[1] 在这种情况下，我们使用了 ChainRules.jl 提供的反向自动微分，以及由 Julia 机器学习软件包 Flux 提供的自动微分系统。Zygote.jl 是与自动微分系统无关的，还可以选择其他方案。

存储中间激活值，因此可以实现内存高效训练。对于地震应用而言，内存效率尤为重要，因为地震模型具有三维体积特性。因此，我们的方法需要能够在这一领域很好地扩展。

为了说明将归一化流作为先验应用于地震反问题的实际用法，我们使用 Compass 模型（Jones 等，2012）的切片训练了一个归一化流。在图4中，我们将归一化流生成的样本与用于训练模型的切片进行比较，后者显示在图3中。尽管仍存在一些不规则之处，该模型已学会了模型的重要定性特征，这些特征将有助于解决反问题。为展示这一方法的实用性，我们在一个全波形反演反问题上测试了我们的先验。由于归一化流先验是独立训练的，因此灵活性很高，可以轻松应用于不同的反问题。

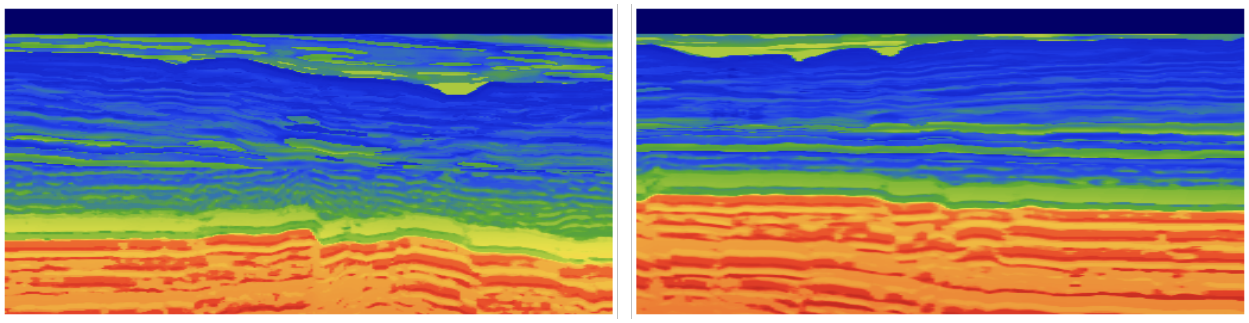


图3：用于训练归一化流先验的 Compass 二维切片示例。

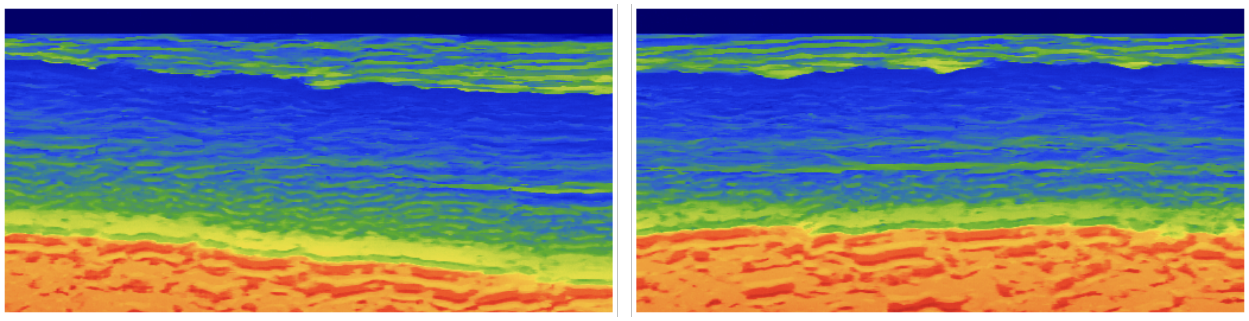


图4：从我们训练好的先验中生成的样本。它们与图 3中的训练样本相似，这表明我们的归一化流学到了有用的先验。

我们的全波形反演实验包括：海底节点、频率低于 4 Hz 时能量为零的雷克子波，以及带宽与无噪声数据相同的加性有色高斯噪声。使用我们的学习先验进行全波形反演时，我们最小化的是

$$\underset{\mathbf{z}}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{F}(\mathcal{G}_{\theta^*}(\mathbf{z}))\mathbf{q} - \mathbf{d}\|_2^2 + \frac{\lambda}{2}\|\mathbf{z}\|_2^2, \tag{2}$$

其中，G_{θ^*} 是一个权重为 θ^* 的预训练归一化流。训练后，归一化流的逆变换将真实的类似 Compass 的地球样本映射到白噪声——即 $\mathrm{G}_{\theta^*}^{-1}(\mathrm{m}) = \mathrm{z} \sim \mathrm{N}(0, I)$。由于归一化流被设计为可逆的，预训练归一化流 G_{θ^*} 作用于高斯噪声 z 后，会生成真实的地球模型样本（见图4）。我们在上式中利用了这一能力，其中 m 中的未知模型参数通过 $\mathrm{G}_{\theta^*}(\mathrm{z})$ 进行重新参数化。正则化项 $\frac{\lambda}{2}\,\mathrm{kzk}_2^2$，会惩罚具有较大 ℓ_2范数的潜变量 z，其中 λ 平衡失配项和正则化项。因此，这种学习型正则化器促使全波形反演结果更有可能成为真实的地球模型（Asim 等，2020）。然而，请注意，

优化流程现在需要同时对物理算子（波动物理，$\mathbf{F}$）和预训练归一化流（$\mathcal{G}_{\theta^*}$）进行求导，只有像我们这样真正的可逆实现，才能在训练和推理时都最大限度地减少内存占用，从而实现可扩展性。

感谢JUDI.jl的 `rrule` 用于 `F` 以及InvertibleNetworks.jl的 `rrule` 用于 `G`，将机器学习与全波形反演相结合变得轻而易举，只需在第 6 行将 `m` 替换为 `G(z)`。现在，最小化公式 2 中的目标函数即可转化为

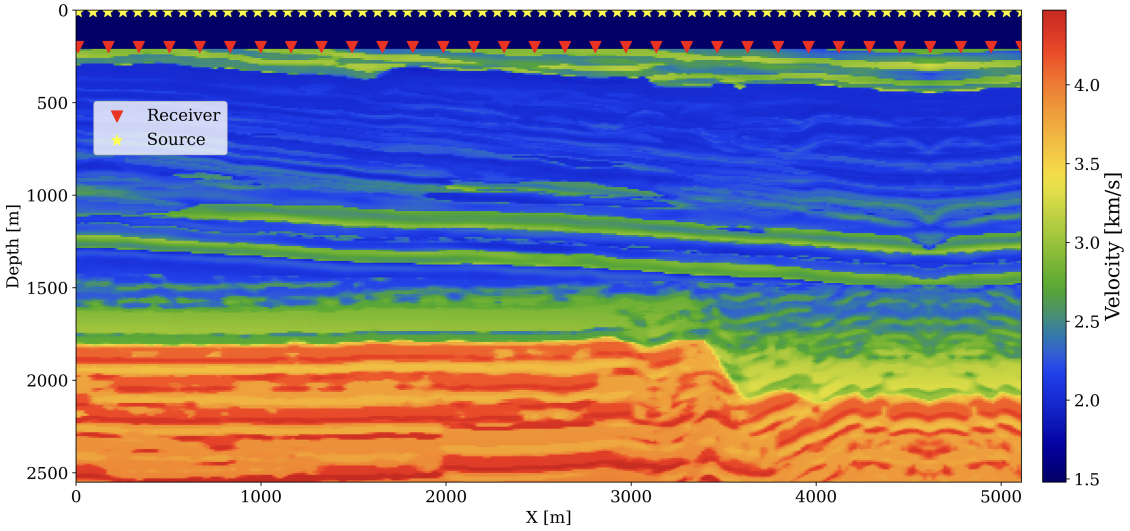
```
# Load the pretrained NF and weights.
G = NetworkGlow(nc, nc_hidden, depth, nscales)
set_params!(G, theta)
# Set up the ADAM optimizer.
opt = ADAM()
# Define the reparameterized loss function including penalty term.
loss(z) = .5f0 * norm(F(G(z)) * q - d)^2f0 + .5f0 * lambda * norm(z)^2f0
# ADAM iterations.
for it = 1:maxiter
    g = gradient(loss, z)[1]      # gradient computation with AD
    update!(opt, z, g)            # update z with ADAM
end
# Convert latent variable to squared slowness.
m = G(z)
```

在图 5中，我们将采用学习先验的全波形反演结果与无正则化全波形反演进行比较。由于我们的先验会将解正则化，使其趋向于符合实际的模型，因此得到的速度估计更接近真实值。

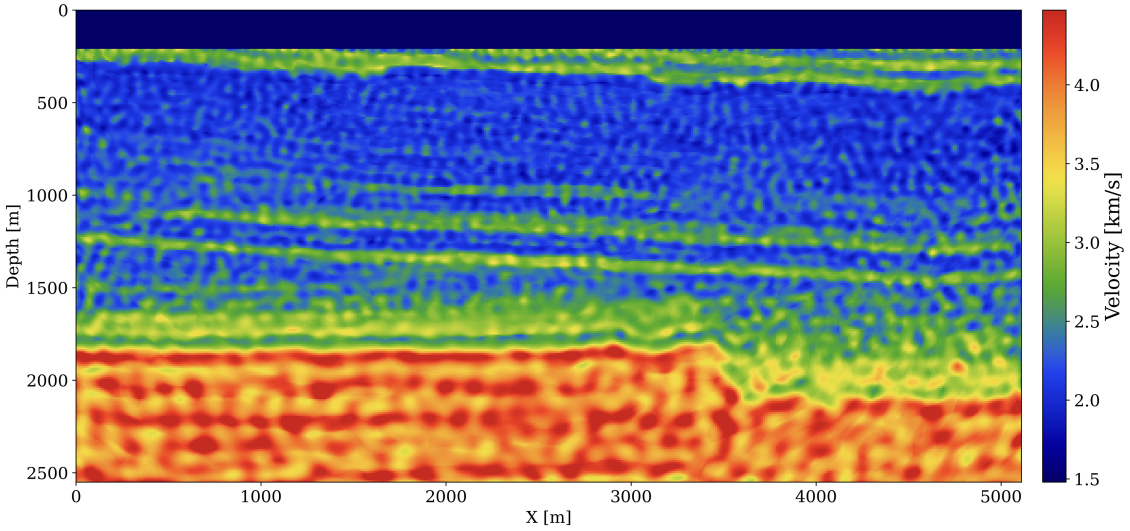
通过这个简单的示例，我们展示了将最先进的波动方程传播算子轻松集成到Julia 的可微编程系统中的能力。通过将这些设计原则应用于端到端反演的其他组件，我们为地下储层中的实际应用设计了一个地震监测框架。

流体流动模拟和渗透率反演

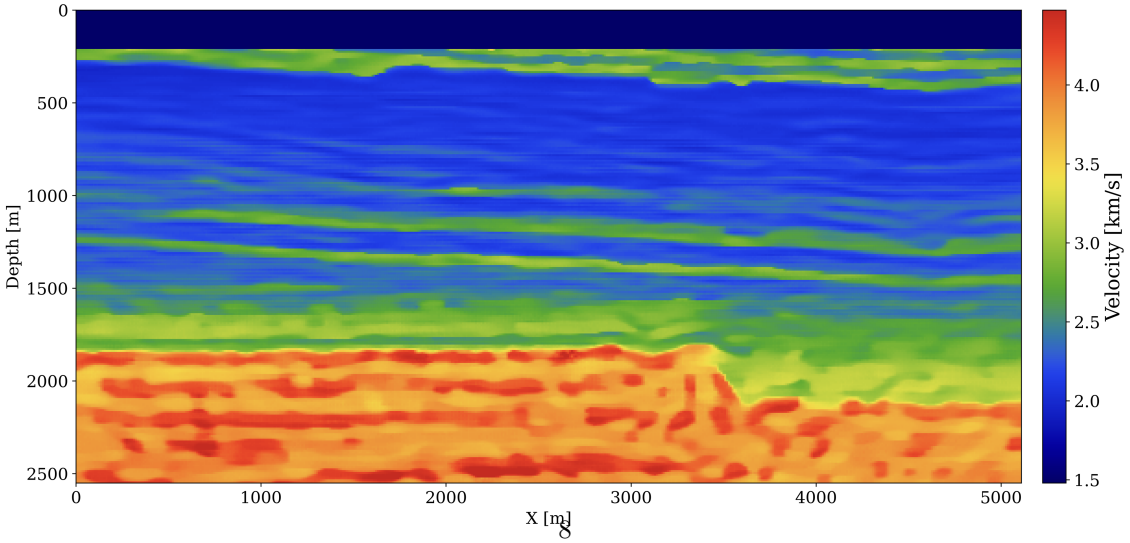
如前所述，我们的目标是根据在CO_2 注入场地采集的时移井间监测数据估计渗透率（参见图 2）。与传统地震成像相比，地质封存的时移监测有所不同，因为其旨在对CO_2 羽状体的时移变化进行成像，同时获得储层流体流动性质的估计。这涉及将波动建模算子与流体流动物理相耦合，以追踪地下的CO_2 羽状体。流体流动物理模型描述了 CO_2 部分取代储层孔隙空间中盐水的缓慢过程，其中涉及求解多相流方程。为此，我们需要使用能够模拟两相（盐水/CO_2）流动的储层模拟软件。虽然现有许多专有和开源储层模拟器，包括MRST（Lie 和Møyner，2021年）、GEOSX（Settgast等人，2022年）和 Open PorousMedia（OPM）（Rasmussen 等，2021年），但很少有模拟器支持对其输出（CO_2 饱和度）相对于其输入（图 **1**中的空间渗透率分布 K）进行微分。我们使用了近期开发的外部Julia 包 JutulDarcy.jl，该包支持达西流，并作为 Jutul.jl 的前端（Møyner 等，2023年），能够提供相对于 $\mathbf{K}$ 的精确雅可比矩阵。 Jutul.jl 是一个用于有限体积离散的隐式求解器，其内部使用自动微分计算雅可比矩阵。它具有可与商业多相流模拟器相媲美的性能和功能集，并能考虑现实效应（例如溶解、相间质量交换、可压缩性、毛细效应）以及残余捕集机制。它还通过地下多相流方程的伴随公式提供精确的灵敏度。为了将该软件包的雅可比矩阵集成到 Julia 的可微编程系统中，我们编写了轻量级的“封装包” JutulDarcyRules.jl。



(a)



(b)



(c)

图 5：在全波形反演中使用我们的归一化流学习先验所得的结果。（*a*）真实值。（*b*）不使用先验的传统FWI。（*c*）使用学习先验得到的我们的全波形反演结果。

(Yin 和 Louboutin 2023) 添加了一个 `rrule` 用于非线性算子$\mathcal{S}(\mathbf{K})$，该算子将渗透率分布$\mathbf{K}$映射为随空间变化的 CO_2 浓度快照$\mathbf{c} = \{\mathbf{c}^i\}_{i=1}^{n_v}$，覆盖 n_v 个监测时间步（参见图1）。添加此 `rrule` 使这些软件包能够与 Julia 自动微分生态系统中的其他软件包互操作。下面展示一个基本示例：给定完整的 CO_2 浓度快照历史记录，使用 ADAM 算法反演地下渗透率：

```
# Generate CO2 concentration.
c = S(K_true)
# Set up ADAM optimizer.
opt = ADAM()
# Define the loss function.
loss(K) = .5f0 * norm(S(K) - c)^2f0
# ADAM iterations.
for it = 1:maxiter
    g = gradient(loss, K)[1]      # gradient computed with AD
    update!(opt, K, g)            # update K with ADAM
end
```

在上述循环的每次迭代中，Julia 的机器学习软件包Flux.jl (Michael Innes 等，2018；Mike Innes, 2018) 会使用前文所述的自定义梯度 `rrule`，调用JutulDarcy.jl中的高性能伴随代码。我们灵活的软件框架还可轻松地以深度学习模型替代数值流体流动模拟器。下一节将介绍分布式傅里叶神经算子（dfno），并探讨这种神经代理模型如何助力我们的反演框架。

傅里叶神经算子代理模型

虽然将多相流建模集成到 Julia 可微编程生态系统中，为开展端到端反演铺平了道路（如下文所述），但流体流动模拟的计算成本很高——考虑到在反演过程中必须多次进行这些模拟，这一问题更加突出。因此，我们转而采用一种数据驱动方法：首先使用模拟样例中的数据对 {K, S(K)} 训练神经算子，使其学习从渗透率模型 K 到相应 CO_2 快照 c :=S(K) 的映射。在承担初始离线训练成本后，该神经代理模型便可在保证可接受精度的同时，作为数值求解器的快速替代方案。傅里叶神经算子（FNO，Z. Li 等，2020）是一种近期提出的神经网络架构，已成功用于地质 CO_2 封存项目中的两相流模拟（Wen 等，2022）。Yin 等人（2022）则独立地使用训练好的傅里叶神经算子替代流体流动模拟，作为端到端反演的一部分，并表明 Julia 的机器学习软件包可以利用 Flux.jl 的反向模式自动微分系统 Zygote.jl（Michael Innes，2018）计算关于渗透率的梯度。训练完成后，只需将 S 替换为 $S_{\mathbf{w}^*}$，其中 $\mathbf{w}^*$ 为预训练傅里叶神经算子的权重，即可根据浓度快照 c 进行上述渗透率反演。借助自动微分系统，关于 K 的梯度会自动计算。因此，在加载训练好的傅里叶神经算子并重新定义算子 S后，上述代码保持完全不变。有关傅里叶神经算子及其训练的实现细节，请参阅 Yin 等人（2022）和 Grady 等人（2022）。

把一切整合起来

作为端到端渗透率反演的最后一步，我们引入一个非线性岩石物理模型，记为$\mathcal{R}$。该模型基于斑块状饱和度模型（Avseth、Mukerji和Mavko，2010），使用Julia 代码将时移 CO_2 饱和度非线性映射为储层内地震属性（纵波速度、$\mathbf{v} = \{\mathbf{v}^i\}_{i=1}^{n_v}$）的降低

```
# Patchy saturation function.# Input: CO2 saturation, velocity, density,
 porosity.# Optional: bulk modulus of mineral, brine, CO2; density of CO2,
 brine.# Output: velocity, density.function Patchy(sw, vp, rho, phi;
bulk_min=36.6f9, bulk_fl1=2.735f9, bulk_fl2=0.125f9,rhow=7f2,
 rhoo=1f3) where T# Relate vp to vs, set modulus properties.
vs = vp ./ sqrt(3f0)bulk_sat1 = rho .* (vp.^2f0 .- 4f0/3f0 .* vs.^2f0)
shear_sat1 = rho .* (vs.^2f0)
# Calculate bulk modulus if filled with 100% CO2.
patch_temp = bulk_sat1 ./ (bulk_min .- bulk_sat1)
.- bulk_fl1 ./ phi ./ (bulk_min .- bulk_fl1)
.+ bulk_fl2 ./ phi ./ (bulk_min .- bulk_fl2)
bulk_sat2 = bulk_min ./ (1f0 ./ patch_temp .+ 1f0)
# Calculate new bulk modulus as weighted harmonic average.
bulk_new = 1f0 / ((1f0 .- sw) ./ (bulk_sat1 .+ 4f0/3f0 * shear_sat1)
+ sw ./ (bulk_sat2 + 4f0/3f0 * shear_sat1)) - 4f0/3f0 * shear_sat1
# Calculate new density and velocity.
rho_new = rho + phi .* sw * (rhow - rhoo)
vp_new = sqrt.((bulk_new .+ 4f0/3f0 * shear_sat1) ./ rho_new)
return vp_new, rho_newend
```

我们将波速的变化映射到时移地震数据中，$\mathbf{d} = \{\mathbf{d}^i\}_{i=1}^{n_v}$，通过块对角地震建模[2] 算子 $\mathcal{F}(\mathbf{v}) = \mathrm{diag}\left(\{\mathbf{F}^i(\mathbf{v}^i)\mathbf{q}^i\}_{i=1}^{n_v}\right)$。在此公式中，单个采集期次的正演算子$\mathbf{F}^i$ 及其对应的源，$\mathbf{q}^i$可以在不同采集期次之间变化。

有了流体流动（替代）求解器 $\mathcal{S}$、岩石物理模块 $\mathcal{R}$ 和波动物理模块 $\mathcal{F}$，以及通过使用$\mathcal{G}_{\theta^*}$进行重新参数化实现的正则化，我们现在可以将所需的端到端反演问题表述为

$$\underset{\mathbf{z}}{\text{minimize}} \quad \frac{1}{2}\|\mathcal{F}\circ\mathcal{R}\circ\mathcal{S}\left(\mathcal{G}_{\theta^*}(\mathbf{z})\right) - \mathbf{d}\|_2^2 + \frac{\lambda}{2}\|\mathbf{z}\|_2^2, \tag{3}$$

其中反演得到的渗透率可通过 $\mathbf{K}^* = \mathcal{G}_{\theta^*}(\mathbf{z}^*)$ 其中 $\mathbf{z}^*$ 是方程 3 的潜空间最小化解。如图 1所示，根据 $\mathcal{F}\circ\mathcal{R}\circ\mathcal{S}$，我们通过组合流体流动、岩石和波动物理，获得非线性端到端映射。相应的 Julia 代码如下

```
# Set up ADAM optimizer.
opt = ADAM()
# Define the reparameterized loss function including penalty term.
loss(z) = .5f0 * norm(F(R(S(G(z)))) - d)^2f0 + .5f0 * lambda * norm(z)^2f0
# ADAM iterations.
for it = 1:maxiter
    g = gradient(loss, z)[1]      # gradient computed by AD
    update!(opt, z, g)            # update z by ADAM
end
```

[2]请注意，我们将该正演建模参数化为压缩波速。

```
# Convert latent variable to permeability.
K = G(z)
```

Yin 等人（2022）将这一端到端反演流程成功应用于一个简单的风格化块状高低渗透率模型，该流程利用了学习型深度先验和预训练 FNO 代理模型。该流程通过采用自动微分，并借助 `rrule` 处理波动物理和流体物理，再结合其固有的自动微分能力，计算方程3中目标函数的梯度；该目标函数综合了流体流动、岩石和波动物理。下文将介绍在一个由真实数据导出的、更贴近现实的场景中应用所提出的端到端反演方法所取得的初步成果（参见图2）。

初步反演结果

尽管 Yin 等人（2022）取得的初步结果令人鼓舞，并显示学习先验具有显著优势，但他们研究中采用的渗透率模型和流体流动模拟过于简单。为在更贴近现实的场景中评估所提出的端到端反演方法，我们考虑图6a所示的渗透率模型。该模型取自图2中 Compass 模型的一个切片（Jones 等，2012）。为在该模型中生成逼真的 CO_2 羽流，我们使用JutulDarcy.jl 进行不混溶可压缩两相流模拟，模拟时长为18年，并绘制了第10、15、16、17和18年的5个快照。这些 CO_2 快照显示在图7的第一行。接下来，给定流体流动模拟结果，我们使用斑块状饱和度模型（Avseth、Mukerji 和 Mavko，2010），将每个 CO_2 浓度快照 $\mathbf{c}^i$, $i = 1 \dots n_v$ 转换为相应的波速模型 $\mathbf{v}^i$, $i = 1 \dots n_v$ 其中 $\mathbf{v}\,\mathrm{R}(\mathbf{c})$。然后，我们使用JUDI.jl 生成合成时移数据 $\mathbf{d}^i$, $i = 1 \dots n_v$，对应每个采集期次。

在反演过程中，对上述合成实验中的前15年时移数据进行反演，$\mathbf{d}^i$, $i = 1 \dots 15$，储层内的渗透率以图6b所示的单一合理值初始化。物理驱动两相流求解器及其学习型神经代理近似在数据上运行25轮后得到的反演结果，分别见图6c 和图6d。这两项结果均通过运行上述代码块200次迭代得到。每个时移采集期次包含960个接收器和32个炮点。为限制波动方程求解的次数，每次迭代仅对随机选取的4个炮点（有放回）计算梯度。虽然这些未采用学习型正则化的结果尚属初步，但可以得出以下观察。首先，渗透率的两项反演结果都呈现出 CO_2的反演锥形。这是预期之中的，因为只有在前15年 CO_2 流经的区域，渗透率才能被反演。其次，反演得到的渗透率呈现出该强非均质模型的变化趋势。第三，正如预期，采用两相流求解器得到的结果细节和连续性更好。这在一定程度上可以解释为：无法保证模型迭代始终处于 FNO 训练所依据的统计分布之内。第四，这一工作流程的实现极大受益于上述软件设计原则。例如，借助抽象，将物理驱动两相流求解器替换为其学习型对应方法变得轻而易举。

尽管反演结果还只是初步的，但对比图$_2$ 7 顶行所示的真实羽流发展情况与图7第三、四行所示的反演模型模拟羽流，可以看出，基于两种反演渗透率模型进行的 18 年 CO_2 模拟是合理的。虽然对过去、当前和预测的 CO浓度估算中缺少某些细节，但与图7第二行所示的渗透率初始模型生成的羽流相比，此次反演取得了显著改进。所介绍工作流程的早期版本可在 Julia 软件包 Seis4CCS.jl 中找到。随着项目逐渐成熟，更新后的工作流程和代码将发布到 GitHub。

仍待解决的挑战

到目前为止，我们希望已经能够让读者相信，使用抽象确实大有裨益。得益于受数学启发的抽象，它们自然而然地带来了模块化和关注点分离，我们

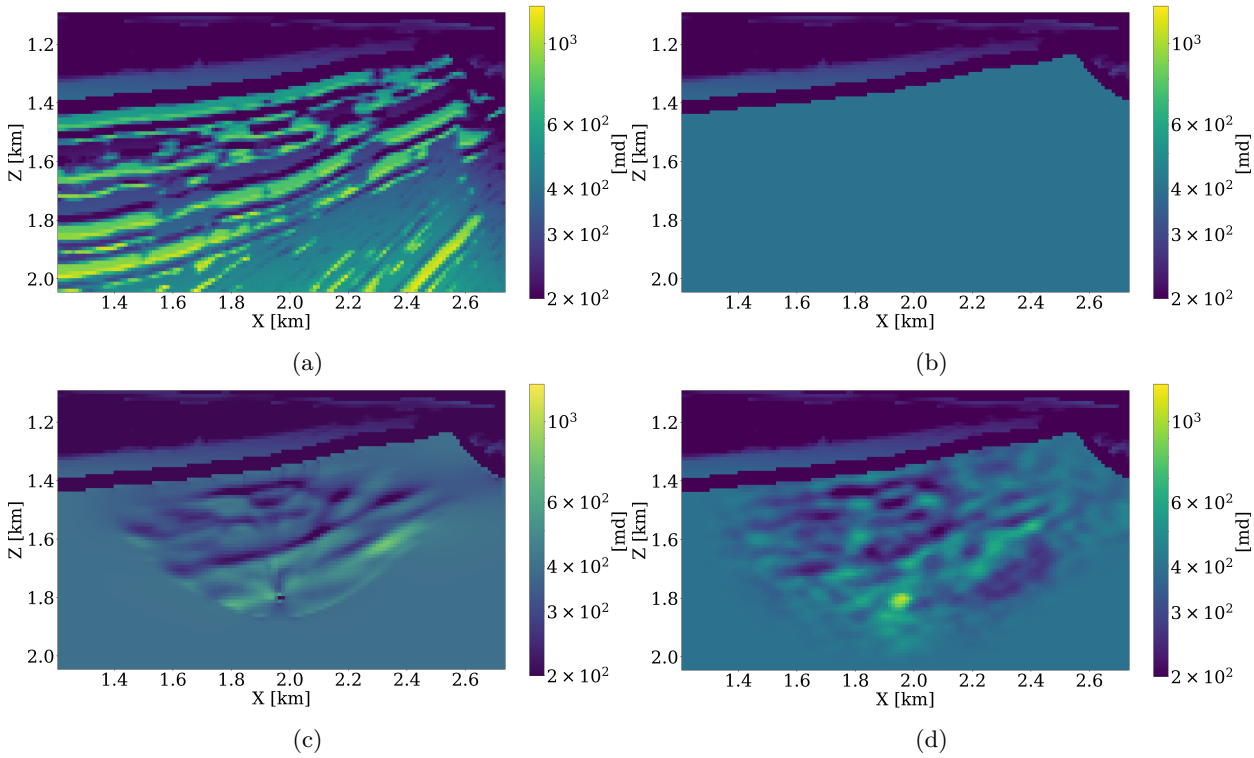


图 6：基于物理和代理流体流动模拟的 15 年时移地震端到端渗透率反演。*(a)* 真实值渗透率。*(b)* 储层中取均一值的初始渗透率。*(c)* 基于物理的反演得到的渗透率。*(d)* 采用神经代理模型近似得到的渗透率。

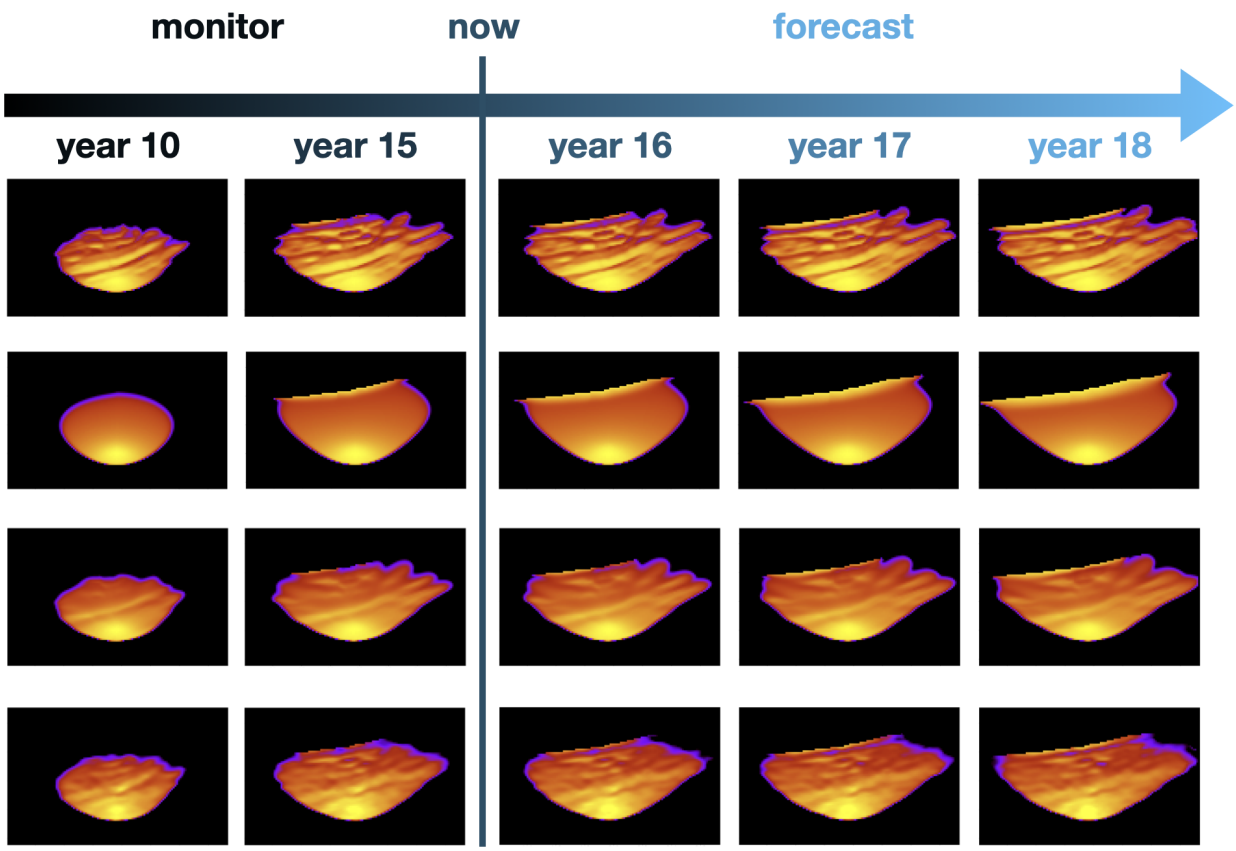


图7：CO_2 羽流估计与预测。前两列是通过地震监测的模拟前15年中第10年和第15年的CO_2 浓度快照。最后三列是第16、17、18年的预测快照，此时没有可用的地震数据。第一行对应于由未见过的真实值渗透率模型模拟得到的真实值CO_2 羽流。第二行包含初始模型中的羽流模拟。第三行和第四行分别包含基于物理的渗透率反演和基于代理模型的渗透率反演所得到的估计和预测CO_2 羽流。

能够加速端到端反演的研发周期。因此，我们创建了一个开发环境，使我们能够纳入机器学习技术。在开发周期相对较晚的时候，它还让我们有机会将最初的二维储层模拟代码替换为由国家实验室开发的功能更强大、特性更完善、达到工业级水平的三维代码。遗憾的是，我们目前仍未能证明可以将这一端到端反演扩展到三维；与此同时，Devito-based propagators and Jutul.jl的流体流动模拟均已在工业规模问题上得到验证。遗憾的是，由于无法获得大规模计算资源，在学术环境中，即使本文介绍的计算工具链是完全可微分的，并且原则上具备扩展能力，我们仍难以在四维合成数据和现场数据上验证所提出的方法。大多数组件都已在现实的三维示例上分别完成测试和验证（Grady 等，2022年；Møyner 和 Bruer，2023年；Mathias Louboutin 和 Herrmann，2022年；Mathias Louboutin 和 Herrmann，2023年），目前正在努力消除根本性的内存瓶颈及其他瓶颈。

规模化归一化流

生成模型（包括归一化流）需要相对较大的训练集和大量用于训练的计算资源。尽管人们一直在努力为更传统的机器学习任务创建训练集，但目前还不存在包含真实三维示例的公共领域训练集。好消息是，与扩散模型（Song 等，2020）相比，归一化流（Rezende 和 Mohamed，2015）的内存占用较小，因此，当训练集和计算资源可用时，训练这类网络将是可行的。在我们的实验室中，我们已经能够在 $256 \times 256 \times 64$ models 上成功训练和评估归一化流。

扩展神经算子规模

自 Z. Li 等，2020 年的开创性论文以来，关于使用傅里叶神经算子作为代理模型来替代昂贵的多相流体流动求解器，以模拟 CO_2 注入（作为地质封存项目的一部分）的研究成果大量涌现（Wen 等，2022、2023）。尽管这种热潮有充分的理由，但当该技术扩展到现实的三维问题时，仍然面临挑战。在这种情况下，必须采取额外措施。例如，通过嵌套傅里叶神经算子，Wen 等（2023）能够将三维域划分为围绕井的小型层次子域；但只有在满足某些假设时，这种方法才可行。由于采用了这种嵌套分解，这些作者避免了三维傅里叶神经算子占用的大量内存，并报告了数量级的加速。鉴于不规则 CO_2 流动（例如泄漏）可能产生的影响，我们尽可能避免对流动行为作出假设，并提出了一种基于网络输入和网络权重域分解的精确分布式傅里叶神经算子（dfno）结构（Grady 等，2022）。通过使用 DistDL（Hewett 和 Grady II，2020）这一支持机器学习中“模型并行”的软件包，我们的 dfno 将输入数据和网络权重划分到多个图形处理器上，使每个分区都能够装入单个图形处理器的内存中。正如 Grady 等（2022）所报告的，我们的工作证明了 dfno 在现实问题和合理规模的训练集（渗透率/CO_2 浓度对）上，对于从斯莱普纳基准模型（Furre 等，2017）派生的渗透率模型，其有效性得到了验证。在 16 个时间步和大小为 $64 \times 118 \times 263$的模型上，相较于 Open Porous Media（Rasmussen 等，2021）的模拟时间（Open Porous Media 是领先的开源储层模拟器之一），我们从自身角度报告了超过 $1300\times$ 的更现实的加速比。这些结果证实了 P. A. Witte 等人（2022）倡导的类似独立方法。尽管我们正与工业合作伙伴及 Extreme Scale Solutions 合作，以进一步改善这些数据，但我们相信，分布式傅里叶神经算子能够以很高的并行效率扩展到三维。

迈向可扩展的开源软件

除了能够复现已发表的结果之外，我们也大力倡导推出可扩展的开源软件，以助力能源转型和应对气候变化。正如其他领域（尤其是机器学习）所观察到的那样，开源软件能够加速创新，而这正是面对重大挑战的行业所需要的特征，

尽管上文介绍了我们实施端到端渗透率反演的经验，但这项工作只是一个正在进行中的项目的阶段性成果。不过，表1 中列出的许多软件组件已经处于高级开发阶段，在很大程度上已准备好进行三维测试，并最终应用于现场数据。例如，我们的所有软件都支持大规模三维模拟和自动微分。此外，我们也已进入支持所有代码使用 GPU 的高级开发阶段。对于关注未来发展的读者，我们还在下文介绍了 Julia 软件包 ParametricOperators.jl，该软件包旨在支持高维并行张量运算，从而为未来原生于 Julia 的分布式傅里叶神经算子实现提供支持。

如果没有其他团队的开源工作，本文所呈现的成果便不可能实现，其中尤为值得一提的是英国伦敦帝国理工学院的研究人员，他们率先推动了Devito的开发，以及挪威 SINTEF 的研究人员。通过将这些软件包集成到 Julia 灵活的可微编程环境中，我们相信，构建一个远胜于各组成部分简单相加的软件环境已指日可待。欢迎读者访问 https://github.com/slimgroup 了解最新进展。

表 1：SLIM的软件技术栈的当前状态。为突出与其他开源项目的协作和积极参与，我们纳入了外部软件包（以 *表示），以及这些软件包如何集成到我们的软件框架中。

软件包	3D	GPU	AD	并行性
Devito*	yes	yes	no	通过 MPI 进行域分解， 通过 OpenMP 实现多线程
JUDI.jl	yes	yes	yes	通过 OpenMP 实现多线程、任务并行 并行
JUDI4Cloud.jl	yes	yes	yes	通过 OpenMP 实现多线程、任务并行 并行
InvertibleNetworks.jl	yes	yes	yes	Julia 原生多线程
dfno	yes	yes	yes	通过 MPI 进行域分解
Jutul.jl*	yes	soon	yes	Julia 原生多线程
JutulDarcyRules.jl	yes	soon	yes	Julia 原生多线程
Seis4CCS.jl	yes	yes	yes	Julia 原生多线程
ParametricOperators.jl	yes	yes	yes	通过 MPI 进行域分解， Julia 原生多线程

结论

在这项工作中，我们为地球物理反问题和机器学习引入了一个软件框架，为大规模研究与开发提供了可扩展、可移植且可互操作的环境。我们展示了，通过精心选择的设计原则，可以创建具有数学启发式抽象的软件，在不牺牲性能的情况下，自然实现所需的模块化和关注点分离。我们通过结合Devito的波传播算子自动代码生成能力与 Julia现代化、高性能且可扩展的编程能力（包括可微编程）来实现这一点。得益于这些特性，我们能够快速实现一个原则上可扩展至三维的渗透率反演原型，用于从时移交叉井地震数据中进行反演。除采用适当的抽象之外，我们解决这一相对复杂的多物理场问题的方法还广泛依赖于 Julia 内置的算法微分能力，并辅以针对波动与流体流动的物理过程以及机器学习组件的高性能辅助导数。基于这些设计选择，我们开发了一个敏捷且相对易于维护的紧凑型软件技术栈，通过数学启发式抽象、现代编程实践和自动代码生成的结合，将底层代码隐藏起来。

代码和数据可用性

我们的软件框架由注册的 Julia软件包组成，均可在 SLIMGitHub 页面（https://github.com/slimgroup）上找到。本文介绍的软件包均为开源软件，并根据 MIT 许可证发布，供社区使用。

致谢

本研究在佐治亚研究联盟和 ML4Seismic 中心产业合作伙伴的支持下开展。作者感谢 Henryk Modzelewski（不列颠哥伦比亚大学）和 Rishi Khan（Extreme Scale Solutions）提供的建设性讨论。本研究还获得了美国国家科学基金会项目（OAC 2203821）和美国能源部项目（编号：DE-SC0021515）的部分资助。

参考文献

Alnaes、M. S.、J. Blechta、J. Hake、A. Johansson、B. Kehlet、A. Logg、C. Richardson、J. Ring、M. E. Rognes 和 G. N. Wells。2015年。“FEniCS 项目 1.5 版。”《数值软件档案》3。https://doi.org/10.11588/ans.2015.100.20553。Asim、Muhammad、Max Daniels、Oscar Leong、Ali Ahmed 和 Paul Hand。2020年。“用于反问题的可逆生成模型：减轻表示误差和数据集偏差。”载于《第37届国际机器学习会议论文集》，119：399–409。《机器学习研究论文集》。PMLR。http://proceedings.mlr.press/v119/asim20a.html。Avseth、Per、Tapan Mukerji 和 Gary Mavko。2010年。《定量地震解释：应用岩石物理工具降低解释风险》。剑桥大学出版社。Berg、Ewout van den 和 Michael P. Friedlander。2009年。“Spot：一款用于 MATLAB 的线性算子工具箱。”不列颠哥伦比亚大学：SCAIM 研讨会；SCAIM 研讨会。Bezanson、Jeff、Alan Edelman、Stefan Karpinski 和 Viral B Shah。2017年。“Julia：数值计算的新方法。”《SIAM Review》59（1）：65–98。https://doi.org/10.1137/141000671。Dinh、Laurent、Jascha Sohl-Dickstein 和 Samy Bengio。2016年。“使用 Real NVP 进行密度估计。”载于国际学习表征会议，ICLR。http://arxiv.org/abs/1605.08803。Fomel、Sergey、Paul Sava、Ioan Vlad、Yang Liu 和 Vladimir Bashkardin。2013年。“Madagascar：用于多维数据分析和可复现计算实验的开源软件项目。”Journal of Open Research Software 1（1）。Furre、Anne-Kari、Ola Eiken、Håvard Alnes、Jonas Nesland Vevatne 和 Anders Fredrik Kiær。2017年。“斯莱普纳二氧化碳注入监测 20 年。”《能源程序》114：3916–26。https://doi.org/https: //doi.org/10.1016/j.egypro.2017.03.1523。Grady、Thomas J.、Infinoid 和 Mathias Louboutin。2022年。Slimgroup/Dfno：Optimal Comm（版本 0.3）。Zenodo。https://doi.org/10.5281/zenodo.6981516。Grady、Thomas J.、Rishi Khan、Mathias Louboutin、Ziyi Yin、Philipp A. Witte、Ranveer Chandra、Russell J. Hewett 和 Felix J. Herrmann。2022年。“作为大规模参数化偏微分方程学习型代理模型的模型并行傅里叶神经算子。”CoRR。https://doi.org/10.48550/ARXIV.2204.01205。Herrmann、Felix J.、Ali Siahkoohi 和 Gabrio Rizzuti。2019年。“带约束和不确定性量化的学习成像。”https://arxiv.org/pdf/1909.06473.pdf。Hewett、Russell J 和 Thomas J Grady II。2020年。“深度学习中模型并行的一种线性代数方法。”arXiv 预印本 arXiv:2006.03108。Innes、Michael。2018年。“不要展开伴随算子：对静态单赋值形式程序求导。”CoRR abs/1810.07951。http://arxiv.org/abs/1810.07951。Innes、Michael、Elliot Saba、Keno Fischer、Dhairya Gandhi、Marco Concetto Rudilosso、Neethu Mariya Joy、Tejan Karmali、Avik Pal 和 Viral Shah。2018年。“使用 Flux 进行时尚建模。”CoRR abs/1811.01457。https://arxiv.org/abs/1811.01457。Innes、Mike。2018年。“Flux：使用 Julia 实现优雅的机器学习。”Journal of Open Source Software。https://doi.org/10.21105/joss.00602。

Innes、Mike、Alan Edelman、Keno Fischer、Chris Rackauckas、Elliot Saba、Viral B Shah 和 Will Tebbutt。2019年。“连接机器学习与科学计算的可微编程系统。”arXiv 预印本 arXiv:1907.07587。Jones、CE、JA Edgar、JI Selvage 和 H Crook。2012年。“构建复杂合成模型，以评估采集几何和速度反演技术。”载于第74届欧洲地球科学家与工程师协会会议暨展览会（含 EUROPEC 2012），cp–293。欧洲地球科学家与工程师协会。Lensink、Keegan、Henryk Modzelewski、Mathias Louboutin、yzhang3198 和 Ziyi (Francis) Yin。2023年。Slimgroup/SegyIO.jl：V0.8.3（版本 v0.8.3）。Zenodo。https://doi.org/10.5281/zenodo.7502671。Li、Dongzhuo、Kailai Xu、Jerry M Harris 和 Eric Darve。2020年。“利用侵入式自动微分进行地下流动问题的耦合时移全波形反演。”《水资源研究》56（8）：e2019WR027032。Li、Zongyi、Nikola Kovachki、Kamyar Azizzadenesheli、Burigede Liu、Kaushik Bhattacharya、Andrew Stuart 和 Anima Anandkumar。2020年。“用于参数化偏微分方程的傅里叶神经算子。”https://arxiv.org/abs/2010.08895。Lie、Knut-Andreas 和 Olav Møyner。2021年。《使用 MATLAB 储层模拟工具箱进行高级建模》。剑桥大学出版社。Lin、Tim T. Y. 和 Felix J. Herrmann。2015年。“SLIM 的学生主导型高性能计算环境。”https: // slim.gatech.edu/Publications/Public/Conferences/IIPFWI/lin2015IIPFWIsdh/lin2015IIPFWIsdh_pres.pdf。Louboutin、Mathias 和 F Herrmann。2022年。“利用随机迹估计在图形处理器上实现基于波动的反演。”载于第83届欧洲地球科学家与工程师协会年会暨展览会，2022：1–5。1。EAGE Publications BV。Louboutin、Mathias 和 Felix J. Herrmann。2023年。“利用随机迹估计在图形处理器上实现大规模基于波动的反演。” https://slim.gatech.edu/Publications/Public/Submitted/2023/louboutin2023rte/paper.html。Louboutin、Mathias、Philipp Witte、Ali Siahkoohi、Gabrio Rizzuti、Ziyi Yin、Rafael Orozco 和 Felix J. Herrmann。2022年。“通过软件抽象加速可扩展计算地球物理学的创新。”载于第二届应用地球科学与能源国际会议，1482–86。https: //doi.org/10.1190/image2022-3750561.1。Louboutin、Mathias、Philipp Witte、Ziyi Yin、Henryk Modzelewski、Kerim、Carlos da Costa 和 Peterson Nogueira。2023年。Slimgroup/JUDI.jl：V3.2.3（版本 v3.2.3）。Zenodo。https://doi.org/10.5281/zenodo.7785440。Louboutin、Mathias、Ziyi Yin 和 Felix J. Herrmann。2022a。Slimgroup/JUDI4Cloud.jl：首次公开发布（版本 v0.2.1）。Zenodo。https://doi.org/10.5281/zenodo.6386831。———。2022b。Slimgroup/SlimOptim.jl：V0.2.0（版本 v0.2.0）。Zenodo。https://doi.org/10.5281/zenodo.701 9463。Louboutin、M.、M. Lange、F. Luporini、N. Kukreja、P. A. Witte、F. J. Herrmann、P. Velesko 和 G. J. Gorman。2019年。“Devito（V3.1.0）：用于有限差分和地球物理勘探的嵌入式领域专用语言。”《地球科学模型开发》12（3）：1165–87。https://doi.org/10.5194/gmd-12-1165- 2019。Luporini、Fabio、Mathias Louboutin、Michael Lange、Navjot Kukreja、Philipp Witte、Jan Hückelheim、Charles Yount、Paul H. J. Kelly、Felix J. Herrmann 和 Gerard J. Gorman。2020年。“Devito（一种自动化模板计算系统）的架构与性能。”《ACM数学软件汇刊》46（1）。https://doi.org/10.1145/3374916。Meurer、Aaron、Christopher P. Smith、Mateusz Paprocki、Ondřej Čertík、Sergey B. Kirpichev、Matthew Rocklin、AMiT Kumar 等。2017年。“SymPy：Python 中的符号计算。”PeerJ Computer Science 3（一月）：e103。https://doi.org/10.7717/peerj-cs.103。Modzelewski、Henryk、Mathias Louboutin、Ziyi Yin、Daniel Karrasch 和 Rafael Orozco。2023年。Slim-group/JOLI.jl：V0.8.5（版本 v0.8.5）。Zenodo。https://doi.org/10.5281/zenodo.7752660。Møyner、Olav 和 Grant Bruer。2023年。Sintefmath/JutulDarcy.jl：V0.2.2（版本 v0.2.2）。Zenodo。https: //doi.org/10.5281/zenodo.7775738。Møyner、Olav、Martin Johnsrud、Halvor Møll Nilsen、Xavier Raynaud、Kjetil Olsen Lye 和 Ziyi Yin。2023年。

Sintefmath/Jutul.jl：V0.2.5（版本 v0.2.5）。Zenodo。https://doi.org/10.5281/zenodo.7775759。Orozco, Rafael, Mathias Louboutin, Ali Siahkoohi, Gabrio Rizzuti, Tristan van Leeuwen, and Felix J. Herrmann。2023年。“用于经颅超声的摊销归一化流与不确定性量化”，3月。https://openreview.net/forum?id=LoJG-lUIlk。Orozco, Rafael, Ali Siahkoohi, Gabrio Rizzuti, Tristan van Leeuwen, and Felix J. Herrmann。2021年。“利用归一化流条件先验进行光声成像。”https://openreview.net/forum?id=wo i1OTvROO1。———。2023年。“伴随算子助力快速、摊销式机器学习贝叶斯不确定性量化。”载于《医学影像2023：图像处理》，Olivier Colliot 和 Ivana Išgum 编，12464：124641L。国际光学与光子学学会；SPIE。https://doi.org/10.1117/12.2651691。Padula, Anthony D., Shannon D. Scott, and William W. Symes。2009年。“用于无坐标线性代数和优化算法抽象表达的软件框架。”《ACM数学软件汇刊》36（2）。https://doi.org/ricevector。Peters, Bas, and Felix J Herrmann。2019年。“凸集与非凸集交集投影的算法与软件及其在反问题中的应用。”arXiv 预印本 arXiv:1902.09699。Peters, Bas, Mathias Louboutin, and Henryk Modzelewski。2022年。Slimgroup/SetIntersectionProjection.jl：V0.2.4（版本 v0.2.4）。Zenodo。https://doi.org/10.5281/zenodo.7257913。Rasmussen, Atgeirr Flø, Tor Harald Sandve, Kai Bao, Andreas Lauser, Joakim Hove, Bård Skaflestad, Robert Klöfkorn 等。2021年。“开放多孔介质流体油藏模拟器。”《计算机与应用数学》81：159–85。Rathgeber, Florian, David A. Ham, Lawrence Mitchell, Michael Lange, Fabio Luporini, Andrew T. T. Mcrae, Gheorghe-Teodor Bercea, Graham R. Markall, and Paul H. J. Kelly。2016年。“Firedrake：通过组合抽象实现有限元法自动化。”《ACM数学软件汇刊》43（3）。https: //doi.org/10.1145/2998441。Ravasi, Matteo, and Ivan Vasconcelos。2020年。“PyLops——用于可扩展代数和优化的线性算子 Python 库。”SoftwareX 11：100361。https://doi.org/https://doi.org/10.1016/j.softx.2019.100361。Rezende, Danilo, and Shakir Mohamed。2015年。“利用归一化流进行变分推断。”载于《机器学习研究论文集》37：1530–38。《机器学习研究论文集》。PMLR。http: //proceedings.mlr.press/v37/rezende15.html。Ringrose, Philip。2020年。《如何在地下储存二氧化碳：早期碳捕集与封存项目的经验》。Springer。Rizzuti, Gabrio, Mathias Louboutin, Rongrong Wang, and Felix J. Herrmann。2021年。“大规模地震反演的波场重建反演双重表述。”《地球物理学》86（6）：1ND–Z3。https://doi.org/10.1190/geo2020-0743.1。Rizzuti, Gabrio, Ali Siahkoohi, Philipp A. Witte, and Felix J. Herrmann。2020年。“通过深度可逆网络参数化不确定性：在油藏表征中的应用。”载于第90届年度国际会议，1541–45。勘探地球物理学家协会；扩展摘要。https://doi.org/10.1190/se gam2020-3428150.1。Rossum, Guido van, and Fred L. Drake。2009年。《Python 3 参考手册》。加利福尼亚州斯科茨谷：CreateSpace。Settgast, Randolph Richard, Benjamin Curtice Corbett, Sergey Klevtsov, Francois Hamon, Christopher Sherman, Matteo Cusini, Thomas Gazzola 等。2022年。“GEOSX。”https://doi.org/10.5281/zenodo.71510 32。Siahkoohi, Ali, and Felix J. Herrmann。2021年。“通过示例学习：利用归一化流实现快速、可靠性感知的地震成像。”载于首届应用地球科学与能源国际会议，1580–85。勘探地球物理学家协会；扩展摘要。https://doi.org/10.1190/segam2021-3581836.1。Siahkoohi, Ali, Gabrio Rizzuti, and Felix J. Herrmann。2020a年。“基于深度学习的地震成像与不确定性量化贝叶斯方法。”载于第82届EAGE会议暨展览。扩展摘要。https://doi.org/10.3997/2214-4609.202010770。———。2020b年。“成像与自动层位追踪中的不确定性量化——一种基于贝叶斯深度先验的方法。”载于第90届年度国际会议，1636–40。勘探地球物理学家协会；扩展摘要。https://doi.org/10.1190/segam2020-3417560.1。———。2020c年。“用于地震成像的弱深度先验。”载于第90届年度国际会议，2998–3002。

勘探地球物理学家协会；扩展摘要。https://doi.org/10.1190/segam2020-3417568.1。———。2022年。“面向任务的地震成像深度贝叶斯推断。”《地球物理学》87（5）：S281–302。https://doi.org/10.1190/geo2021-0666.1。Siahkoohi, Ali, Gabrio Rizzuti, Mathias Louboutin, Philipp Witte, and Felix J. Herrmann。2021年。“用于反问题变分推断的归一化流预条件训练。”https: // openreview.net/pdf?id=P9m1sMaNQ8T。Siahkoohi, Ali, Gabrio Rizzuti, Rafael Orozco, and Felix J. Herrmann。2023年。“基于物理的潜在分布校正实现可靠的摊销变分推断。”《地球物理学》88（3）。https://doi.org/10.119 0/geo2022-0472.1。Silva, Curt Da, and Felix J. Herrmann。2019年。“用于非线性反问题的统一二维/三维大规模软件环境。”《ACM数学软件汇刊》。https://slim.gatech.edu/Publications/Public/Journals/ACMTOMS/2019/dasilva2017uls/dasilva2017uls.html。Song, Yang, J ascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole。2020年。“通过随机微分方程进行基于得分的生成建模。”arXiv 预印本 arXiv:2011.13456。Sun, Dong, and William W Symes。2010年。“IWAVE伴随状态法的实现。”技术报告 10-06，莱斯大学计算与应用数学系，美国得克萨斯州休斯顿。https://pdfs.semanticscholar.org/6c17/cfe41b76f6b745c435891ea6ba6f4e2c2dbf.pdf。Washbourne, John, Sam Kaplan, Miguel Merino, Uwe Albertin, Anusha Sekar, Chris Manuel, Scott Mishra, Matthew Chenette, and Alex Loddoch。2021年。“Chevron成像与反演优化框架（COFII）——一个开源、云友好的 Julia 语言地震建模与反演框架。”载于首届应用地球科学与能源国际会议扩展摘要，792–96。https://doi.org/10.1190/segam2021-3594362.1。Wen, Gege, Zongyi Li, Kamyar Azizzadenesheli, Anima Anandkumar, and Sally M Benson。2022年。“U-FNO——一种基于增强型傅里叶神经算子的多相流深度学习模型。”《水资源进展》163：104180。Wen, Gege, Zongyi Li, Qirui Long, Kamyar Azizzadenesheli, Anima Anandkumar, and Sally Benson。2023年。“利用嵌套傅里叶神经算子实时、高分辨率预测二氧化碳地质封存。”《能源与环境科学》。White, Frames Catherine, Michael Abbott, Miha Zgubic, Jarrett Revels, Seth Axen, Alex Arslan, Simeon Schaub 等。2023年。“JuliaDiff/ChainRules.jl：V1.47.0。”Zenodo。https://doi.org/10.5281/zenodo.7628788。White, Frames Catherine, Miha Zgubic, Michael Abbott, Jarrett Revels, Nick Robinson, Alex Arslan, David Widmann 等。2022年。“JuliaDiff/ChainRulesCore.jl：V1.15.6。”Zenodo。https://doi.org/10.5281/zenodo .7107911。Witte, Philipp A., Russell J. Hewett, Kumar Saurabh, AmirHossein Sojoodi, and Ranveer Chandra。2022年。“SciAI4Industry——利用深度学习解决工业规模问题中的偏微分方程。”https://arxiv.org/abs/ 2211.12709。Witte, Philipp A., Mathias Louboutin, Navjot Kukreja, Fabio Luporini, Michael Lange, Gerard J. Gorman, and Felix J. Herrmann。2019年。“用于 Julia 中地震反演算法符号化实现的大规模框架。”《地球物理学》84（3）：F57–71。https://doi.org/10.1190/geo2018-0174.1。Witte, Philipp A., Mathias Louboutin, Fabio Luporini, Gerard J. Gorman, and Felix J. Herrmann。2019年。“利用即时傅里叶变换进行压缩最小二乘偏移成像。”《地球物理学》84（5）：R655–72。https://doi.org/10.1190/geo2018-0490.1。Witte, Philipp, Mathias Louboutin, Rafael Orozco, Gabrio Rizzuti, Ali Siahkoohi, Felix Herrmann, Bas Peters, Páll Haraldsson, and Ziyi Yin。2023年。Slimgroup/InvertibleNetworks.jl：V2.2.4（版本 v2.2.4）。Zenodo。https://doi.org/10.5281/zenodo.7693048。Yang, Mengmeng, Zhilong Fang, Philipp A. Witte, and Felix J. Herrmann。2020年。“结合震源估计的时域稀疏性促进最小二乘逆时偏移。”《地球物理勘探》68（9）：2697–2711。https://doi.org/10.1111/1365-2478.13021。Yin, Ziyi, Huseyin Tuna Erdinc, Abhinav Prakash Gahlot, Mathias Louboutin, and Felix J. Herrmann。2023年。“利用高分辨率时移地震数据对地质碳封存风险进行去风险化，以解释泄漏

检测。”《*The Leading Edge*》 42 (1)：69–76。 https://doi.org/10.1190/tle42010069.1。Yin, Ziyi 和 Mathias Louboutin。2023年。 *Slimgroup/JutulDarcyRules.jl*：*V0.2.4* （版本 v0.2.4）。Zenodo。 https://doi.org/10.5281/zenodo.7762154。Yin, Ziyi、Mathias Louboutin 和 Felix J. Herrmann。2021年。“利用联合恢复模型对碳封存与固存进行压缩时移地震监测。” https://doi.org/10.1190/segam2021-3569087.1。Yin, Ziyi、Rafael Orozco、Philipp A. Witte、Mathias Louboutin、Gabrio Rizzuti 和 Felix J. Herrmann。2020。“扩展源成像——地震与医学成像的统一框架。” https://doi.org/10.1190/segam2020-3426999.1。Yin, Ziyi 、 Ali Siahkoohi 、 Mathias Louboutin 和 Felix J. Herrmann。2022年。“利用傅里叶神经算子进行碳封存监测与预测的学习型耦合反演。” https://doi.org/10.1190/image2022-3722848.1。Zhang, Xin 和 Andrew Curtis。2020。“使用变分推断方法进行地震层析成像。”《地球物理研究杂志：固体地球》 125 (4)：e2019JB018589。———。2021年。“使用可逆神经网络进行贝叶斯地球物理反演。”《地球物理研究杂志：固体地球》 126 (7)：e2021JB022320。Zhao, Xuebin、Andrew Curtis 和 Xin Zhang。2021年。“使用归一化流进行贝叶斯地震层析成像。”《国际地球物理学杂志》 228 (1)：213–39。 https://doi.org/10.1093/gji/ggab298。