arXiv:2611.31865v1 [cond-mat.mtrl-sci] 24 Nov 2026

加速等变原子间势通过分子动力学实现径向制表与内核融合

Jaekyun Yoo 和 Suyeon Ju

独立研究者

电子邮箱： worbs3237@gmail.com；suyeonju224@gmail.com

摘要

等变机器学习原子间势（MLIPs）能够达到接近量子化学的精度，但其计算成本和 GPU 内存占用限制了长空间尺度和长时间尺度上的分子动力学（MD）模拟。即使张量积（TP）得到加速，NequIP 卷积中的逐边径向权重的计算与存储仍然是这两个瓶颈的主要来源。我们提出 JET-MLIPs（用于 MLIP 模拟的联合内核等变卷积与制表径向权重），这是一种适用于预训练 NequIP 系列势的 GPU 后端，无需重新训练。JET-MLIPs 以分段三次 Hermite 表替代径向多层感知机（MLP），并将表查找、球谐函数、TP 和邻居聚合融合到单个内核中，从而在力评估过程中消除逐边径向权重和 MLP 中间结果的存储。在单个 H200 GPU 上，针对 5632 个原子的液态电解质体系，使用 SevenNet、MACE、NequIP-OAM、Nequix 和 EquFlash 发布的 15 个势进行测试时，LAMMPS 分子动力学相较于各系列现有的 TP 加速后端，实现了 1.5–5.9$\times$ 的步时间加速比和 2.4–24$\times$ 的峰值内存减少，同时保持单精度（fp32）精度和动力学稳定性。这些改进使得在单个 GPU 上对约 10^4 个原子组成的体系进行多纳秒输运性质模拟成为可能，并扩大了在广泛可用的 GPU 上开展大规模等变 MLIP 分子动力学模拟的可及性。

1 引言

机器学习原子间势（MLIPs）使原子模拟的空间和时间尺度远远超出了量子化学计算所能达到的范围。[1–4] 图神经网络（GNNs）通过在原子图上进行消息传递来学习原子表示，[5,6] 在此基础上，该领域日益转向通用机器学习原子间势，使其能够直接驱动生产级分子动力学（MD），而无需针对特定任务的训练数据。[7–10] 等变消息传递由 NequIP 首创，是这一进展的核心，[11] 其中，内部特征携带方向信息，并随原子结构一致地旋转。如今，许多现代架构和通用势都采用这种设计，以实现接近量子化学的精度。[12,13]

尽管取得了这些进展，基于机器学习间势的分子动力学对于许多实际应用中的材料性质仍然受到限制，因为这些性质需要扩展的空间和时间尺度。预测扩散系数、离子电导率和黏度等输运性质，需要数纳秒至数十纳秒的轨迹，[14] 对应于 10^5–10^8 次模型评估。每次评估的运行时间和 GPU 内存都随原子数增加而增长，因此设备内存限制了单个图形处理器能够容纳的最大模拟晶胞。这一上限对于需要大型模拟晶胞的体系尤其具有约束性，包括带有晶界的多晶固体、[15]固-固界面[16]以及液态电解质。[14]因此，借助基于机器学习间势的分子动力学达到如此扩展的尺度，需要数千至数万个图形处理器，[17–19] 这是大多数研究人员无法获得的资源。[20] 在实践中，机器学习原子间势研究因此往往局限于静态性质计算或短时轨迹，[21,22] 而大规模、长时间尺度的经典分子动力学仍主要依赖经验原子间势。

优化等变机器学习势的 GPU 执行，以同时降低运行时间和内存占用，是在无需重新训练或牺牲精度的情况下扩展可实现分子动力学规模的直接途径。在 SevenNet、[23] MACE[13] 和 NequIP-OAM[24] 等 NequIP 系列架构中，实现这些优化主要取决于加速共享卷积运算；由于每个

原子都会从其邻居处聚合信息，因此该运算在评估时间和内存使用方面都占据主导地位。在这一卷积过程中，一个称为径向多层感知机（MLP）的小型神经网络将每个原子间距离映射为径向权重，随后张量积（TP）将这些权重与邻居特征及键方向相结合。在 e3nn 中，[25] 作为这些机器学习势底层依赖的 PyTorch 库，TP 主导了卷积运行时间，占 7net-l3i5 推理时间的 89%。[26]在我们的测量中，用 cuEquivariance 替换 e3nn TP 和邻居聚合同样消除了 7net-nano-5 卷积时间的 89–91%。5[27] 以及 7net-0（补充信息图 S9a）。因此，近期的加速工作主要集中于 TP；cuEquivariance、[28] FlashTP[26] 和 OpenEquivariance[29] 提供了专用 GPU 内核，作为这些机器学习势现有的 TP 加速后端。

尽管 TP 已得到加速，但作为卷积另一部分的径向多层感知机却几乎没有得到加速。cuEquivariance、FlashTP 和 OpenEquivariance 都将径向权重作为输入，而这些权重是在其内核之外计算的。然而，据我们的统计，对于每一对原子，径向多层感知机而非 TP 承担了大部分算术运算（见第 2.2 节和 SI 表S1）。将力计算纳入考虑后，径向多层感知机同样对内存有较高需求，因为每一对原子的径向权重和中间结果都必须保留在内存中，直到完成力的计算（SI 图 S9b）。将径向多层感知机融合到内核中可以移除大部分这类内存占用，但其本身并不能缩短步时间，因为多层感知机的算术运算仍然存在。因此，要同时缩短步时间并减少内存占用，就需要一种更廉价的径向权重获取方式。

制表提供了这样一种方法。训练好的模型的径向权重是原子间距离的固定函数，因此可以预先计算一次，存储在表中，并在分子动力学过程中通过插值读取。DP Compress[30] 将这一思路应用于不变量深度势模型，通常可使分子动力学模拟速度提升一个数量级，同时使单个图形处理器可容纳的最大体系规模也以相近幅度提升。然而，其性质精度仅在双精度浮点数（fp64）下得到确认。在单精度浮点数（fp32）下，即使使用更精细的表，误差也没有降低，作者将其归因于有效数字有限。近期，DPA4C[31]

将径向制表扩展到了等变机器学习势，但仅适用于使用其自身架构训练的模型；该架构与其 GPU 内核协同设计。径向制表与内核融合相结合能否扩展到现有的等变机器学习势，包括已发布的 NequIP 系列模型，尚未得到验证。

在此，我们介绍 JET-MLIPs（用于机器学习原子间势模拟的、采用表格化径向权重的联合内核等变卷积），这是一种用于机器学习势推理的 GPU 后端，可利用已发布的 NequIP 系列机器学习原子间势检查点加速分子动力学，无需重新训练。它支持 SevenNet、MACE、NequIP-OAM、Nequix[32] 和 EquFlash[33] 在 fp32 精度下运行，既可用于 LAMMPS[34]，也可通过原子模拟环境（ASE）的计算器接口使用。[35] JET-MLIPs 不会即时计算径向多层感知机，而是在单个融合内核中通过三次 Hermite 插值从表中查找预先计算的径向权重；该内核还会计算球谐函数、张量积和邻居聚合。表查找不会产生多层感知机中间值，因此，力的反向传播无需在全局内存中保留径向权重或隐藏层激活值。

我们评估了涵盖这五个系列的 15 个已发布 JET-MLIPs 模型。我们首先评估数值精度，因为只有在加速的同时保留原始势的预测结果，加速才有意义。具体而言，我们验证了 JET-MLIPs 在 fp32 算术的固有精度限制内复现原始模型；判断依据是现有 fp32 后端的数值变化所设定的精度预算（第 2.4 节）。针对这五个系列，我们进行了单点 fp64 参考值检查、独立的 fp64 表评估，并对代表性模型开展了分子动力学验证。结果表明，相较于 fp32 舍入，径向制表引入的力误差可以忽略不计，同时能量守恒和分子动力学观测量均保持在现有后端的运行间波动范围内（第 3.1 节）。在确保精度的前提下，对于一个含 5632 个原子的电解质体系（未使用 D3 色散校正[36,37]），在单块 NVIDIA H200 GPU 上运行 LAMMPS 分子动力学时，各模型的速度均比所属系列现有的 TP 加速后端快 1.5–5.9× 倍（表 2 和第 3.2 节），并且所有模型的全程峰值 GPU 内存占用均减少了 2.4–24×（第 3.5 节）。我们还将这些加速比分解为以下几项的贡献：

制表、内核融合和 LAMMPS 集成（第 3.3 节），并考察了它们在三种致密固体以及液态电解质中的系统规模扩展特性（第 3.4 节）。最后，我们通过计算液态 $LiPF_6$ 碳酸酯电解质的 Green–Kubo 剪切黏度，以及 Cl/S 无序硫银锗矿型 Li_6PS_5Cl 的锂自扩散和离子电导率，展示了这种加速的实际影响；计算基于单块 GPU 上、包含约 10^4 个原子的多纳秒级轨迹（第 3.6 节）。这些步时间和内存占用的缩减降低了轨迹计算成本，也放宽了对系统规模和 GPU 选择的硬件限制。预计这种后端级加速将使大规模、长时间尺度的机器学习原子间势模拟成为常规操作，从而能够研究以往难以触及的输运和微观结构现象。

2 理论与方法

2.1 NequIP 卷积与力计算

本研究考察的五种机器学习原子间势系列，即 SevenNet、[23] MACE、[13] NequIP-OAM、[24] Nequix[32] 和 EquFlash，[33] 均基于 NequIP 的等变消息传递架构[11]（图 1）。在该架构中，网络嵌入每个原子 i的原子种类 Z_i，通过 T 相互作用层更新原子表征，并对读出的原子能量 E_i 求和，以获得总势能 E（图 1a）。在网络中，每个原子 i 携带等变特征 x_i，这些特征按 m_l 通道组织（以u为索引），对应各个旋转阶数 $l \le l_{\max}$（补充信息表 S1 和 S2）。每个通道表示一个由 $2l+1$ 个分量构成、次数为l 且在空间反演下具有确定奇偶性的特征，类似于标准神经网络中的一个隐藏单元。[25,38] 旋转时，标量特征（$l=0$）保持不变，而更高次数的特征（$l>0$）则随原子结构协调旋转。

在每个交互层中 t，一个预卷积块（例如线性层$_{\text{pre},\ t}$ 见图 1a）将映射

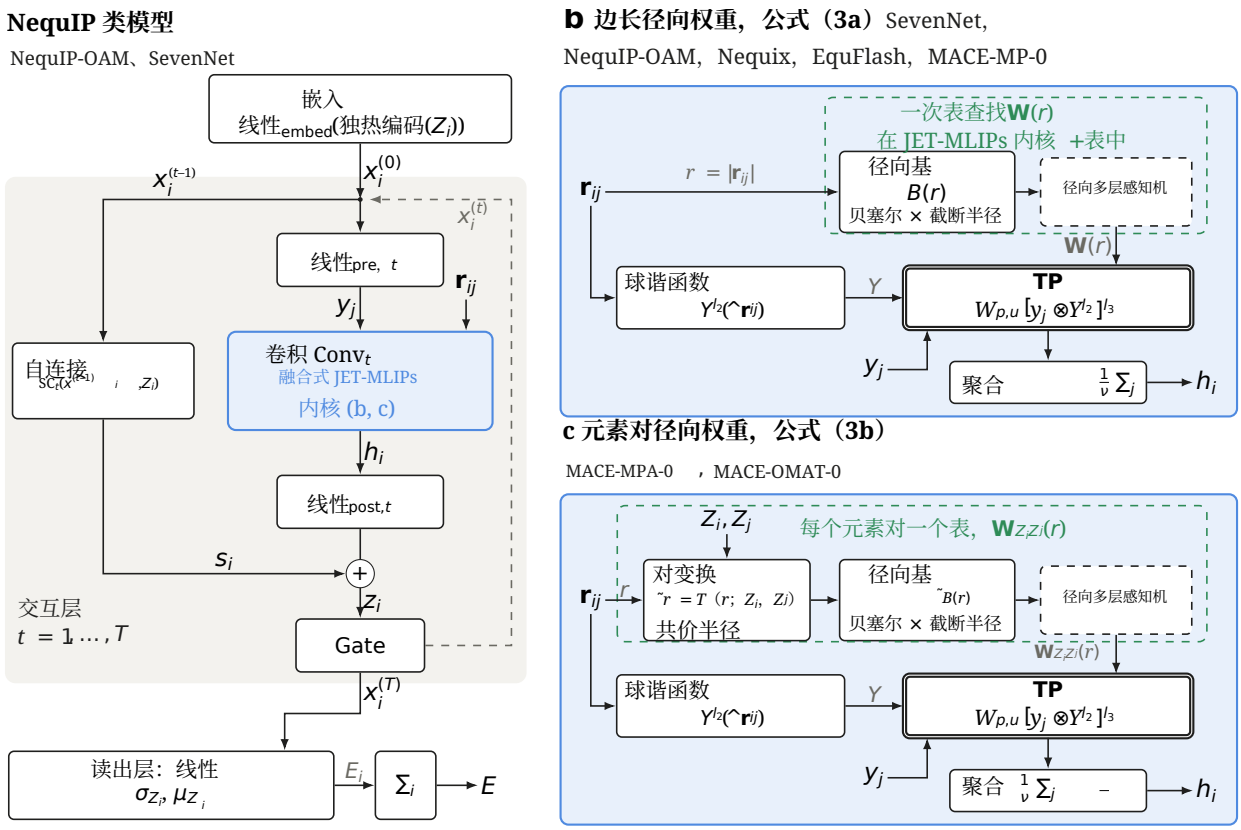


图 1：NequIP 类模型的交互层，以及由 JET-MLIPs 替代的卷积。(a) 嵌入、交互层和原子能量读出，以 NequIP-OAM 和 SevenNet 为例进行说明。两个模型在卷积之外的细节有所不同。NequIP-OAM 在其第一层中省略了自连接，使用单个线性读出层，并将 Ziegler–Biersack–Littmark（ZBL）对项添加到原子能量中。(b) 使用边长径向权重的卷积。(c) 使用元素对依赖的径向权重的卷积，每个元素对使用一个表。径向多层感知机（MLP，虚线轮廓）生成 W，张量积（TP，双线轮廓）将特征与球谐函数耦合。蓝色阴影表示融合的前向与力反向内核，单次表查找取代了绿色虚线框内的操作。

该层的t 输入特征 $x_j^{(t-1)}$，每个邻居 j 到 y_j。NequIP 卷积（图1b、c），

$$h_i = \mathrm{Conv}_t\big(\{y_j, \mathbf{r}_{ij}\}_{j\in\mathcal{N}(i)}\big), \tag{1}$$

随后结合特征 y_j 邻近原子的 $j \in \mathcal{N}(i)$（位于截断半径 r_c 以内）与原子间向量$\mathbf{r}_{ij}$，生成卷积特征 h_i。尽管Conv_t 前后的操作在五个系列中各不相同，但它们都共享这一卷积的架构。图1a 展示了 NequIP-OAM 和 SevenNet 的 NequIP 类型层，层方程见补充信息 S2.1。

对于每条有向边 $j \to i$，其原子间向量为$\mathbf{r}_{ij}$，距离为 $r = |\mathbf{r}_{ij}|$，方程（Conv_t）中的卷积计算1得出

$$B_n(r) = \frac{2}{r_c}\,\frac{\sin(c_n r)}{r}\, f_{\mathrm{cut}}(r), \qquad n = 1, \ldots, N_B, \tag{2}$$

$$\mathbf{W}(r) = \mathrm{MLP}\big(B(r)\big) \in \mathbb{R}^{N_W}, \qquad N_B \to H \to \cdots \to H \to N_W,\ \mathrm{SiLU}, \tag{3a}$$

$$\mathbf{W}(r) = \mathrm{MLP}\big(\tilde{B}(r)\big), \qquad \tilde{B}_n(r) = \frac{2}{r_c}\,\frac{\sin(c_n \tilde{r})}{\tilde{r}}\, f_{\mathrm{cut}}(r), \qquad \tilde{r} = T(r;\, Z_i, Z_j), \tag{3b}$$

$$m_{ij}^{(p,u)} = W_{p,u}(r)\ \ \big[y_j^{(l_1,u)} \otimes Y^{l_2}(\hat{\mathbf{r}}_{ij})\big]^{l_3}, \tag{4}$$

$$h_i = \frac{1}{\nu}\sum_{j\in\mathcal{N}(i)} m_{ij}. \tag{5}$$

公式（2）将原子间距离 r 展开到由频率为 c_n 的 N_B 贝塞尔函数构成的径向基 $B(r) = (B_1, \ldots, B_{N_B})$ 中，再乘以在 r_c 处归零的平滑截断包络函数 $f_{\mathrm{cut}}(r)$。随后，径向多层感知机（公式（3a）和（3b））将基向量依次输入 n_h 个隐藏层，每层有 H 个单元，并通过 Sigmoid 线性单元（SiLU）激活，得到边的 N_W 径向权重 $\mathbf{W}(r) \in \mathbb{R}^{N_W}$。在 SevenNet、NequIP-OAM、Nequix、EquFlash 和 MACE-MP-0 中，多层感知机直接接收 $B(r)$，因此 $\mathbf{W}(r)$ 仅取决于边长 r（公式（3a），图1b）。相比之下，在 MACE-MPA-0 和 MACE-OMAT-0 中，距离变换 T 会先根据（Z_i，Z_j）的共价半径，将贝塞尔函数的自变量重新缩放为 $\tilde{r}$，同时仍在 r 处计算 $f_{\mathrm{cut}}(r)$，从而得到修正后的基 $B(r)$ 以及元素对依赖的径向权重

(式（3b），图 1c）。

公式（4）通过张量积（TP）将角度依赖性引入边消息 $m_{ij}^{(p,u)}$。该张量积将邻居 j 在通道 u 中的 l_1 阶输入特征 $y^{(l_1,u)}$）$_j$ 与单位边向量 $\hat{\mathbf{r}}_{ij} = \mathbf{r}_{ij}/r$ 的实球谐函数 Y^{l_2} 耦合起来。括号 $[\cdot]^{l_3}$ 将它们的张量积 $\otimes$ 投影到满足 $|l_1 - l_2| \leq l_3 \leq l_1 + l_2$、且在空间反演下保持奇偶性的输出阶数 l_3（补充信息图S1a）。每个有效的阶数三元组 $p = (l_1, l_2, l_3)$ 都定义了张量积的一条路径。[25] 卷积前后的线性层会混合同一阶数的通道，而张量积本身则独立作用于每个通道 u。因此，每条路径 p 都会为每个通道 u使用一个标量径向权重 $W_{p,u}(r)$，每条边共有 $N_W = \sum_p m_{l_1}(_p)$ 个径向权重。最后，公式（5）将所有路径和通道上的 $m_{ij}^{(p,u)}$ 汇集为边消息 m_{ij}，对 r_c 内的所有邻居 $j \in \mathcal{N}(i)$ 求和，并除以检查点中存储的归一化常数 ν，从而得到公式（1）的卷积输出 h_i。

每个分子动力学步都需要一次前向传播来评估 E，随后进行一次反向传播以计算原子力 $\mathbf{F} = -\partial E/\partial \mathbf{r}$，因为力是总能量相对于原子位置的负梯度。本文提出的内核取代了式（2）–（5）中的前向传播和力反向传播，包括边嵌入（定义为计算 B_n、f_{cut} 和 Y^{l_2}，由 $\mathbf{r}_{ij}$ 得到）以及每个交互层的卷积。相比之下，模型训练需要对力反向传播求导，涉及网络参数，因此不在本文讨论范围内。

2.2 等变机器学习原子间势的计算瓶颈

在卷积中，径向多层感知机和张量积承担了大部分逐边计算工作：径向多层感知机生成径向权重 $\mathbf{W}(r)$，张量积则使用这些权重。在现代图形处理器上，这些阶段的执行时间不仅取决于算术运算次数，还取决于从全局内存读取和写入的每边中间数据量。在 e3nn 等未融合实现中，[25] 计算

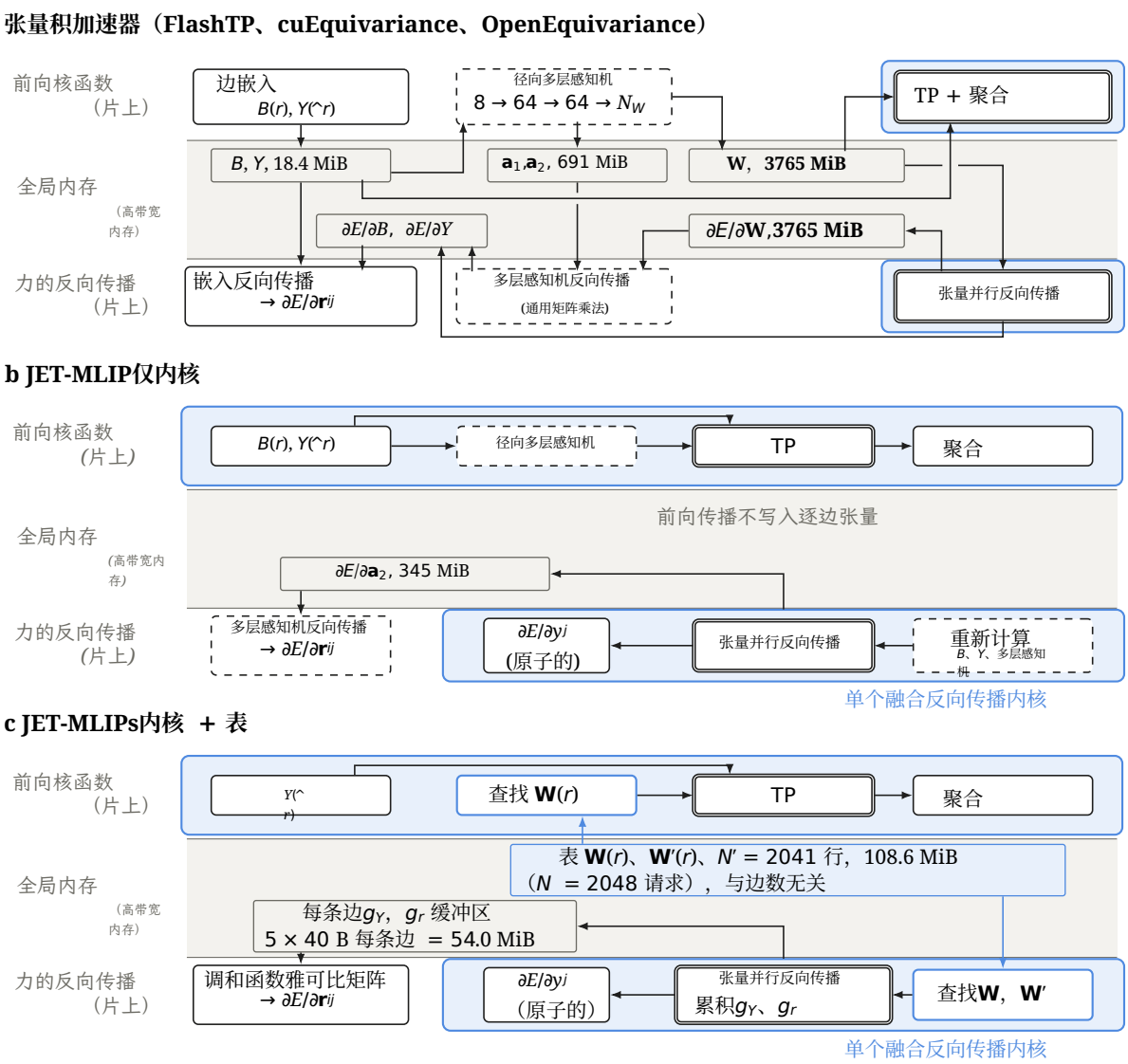


图 2：使用张量积加速器和 JET-MLIPs 时，一个 7net-0 交互层的数据流。（a）张量积加速器将张量积与聚集和聚合（蓝色方框）融合，但会将径向权重$\mathbf{W}$、其梯度 $\partial E/\partial \mathbf{W}$、径向基 B、球谐函数Y 以及 MLP 激活值 $\mathbf{a}_1$, $\mathbf{a}_2$ 写入全局内存。（b）仅融合内核模式将$\mathbf{W}$ 和 $\partial E/\partial \mathbf{W}$ 保存在寄存器中。只有 $\partial E/\partial \mathbf{a}_2$ 会经过全局内存，传递到完成 MLP 反向传播的第二个内核。（c）默认表使用三次厄米查找表替代 MLP，第二个短内核则将相对于球谐函数和边长的缓冲梯度 g_Y 和 g_r转换为 $\partial E/\partial \mathbf{r}_{ij}$。张量大小是在 5632 个原子上对 7net-0 的五个交互层求和得到的，其中 B 和 Y 仅计算一次。灰色方框表示全局内存（HBM，高带宽内存）中的逐边张量，虚线轮廓表示 MLP，双线轮廓表示张量积；力的反向传播从右向左进行。GEMM 表示稠密矩阵乘积。

将邻居收集、TP 和散射作为独立操作[39] 会在全局内存中物化每条边的特征张量和消息张量，而 TP 阶段占据了模型调用时间的 80% 以上[26] （补充信息S4.3）。现有的 TP 加速器，包括 FlashTP、cuEquivariance 和 OpenEquivariance，利用克莱布施–戈尔丹系数的稀疏性，将聚集、TP 和散射步骤融合到单个内核中，从而消除 TP 阶段的内存流量（图 2a）。

然而，即使 TP 已融合，径向权重仍然是主要瓶颈，因为径向多层感知机主导了每条边的算术运算，而 W(r) 则主导了剩余的每条边内存流量。为量化算术成本，我们统计每条边所需的融合乘加（FMA）操作数（$a \cdot b + c$，按单条指令计算）。尽管式（3a、b）中的径向多层感知机仅以标量对距离 r （扩展为 N_B 个基函数值）作为输入，但它必须为每条 TP 路径和每个通道分别生成权重 $W_{p、u}(r)$，因此会产生长度为 N_W 的宽输出向量。将 N_B 个基函数值依次映射至 n_h 个、每个包含 H 个单元的隐藏层，再映射为这些 N_W 个径向权重，需要每层执行一次矩阵–向量乘积；每个矩阵元素对应一次 FMA，因此输入层需要 $N_B H$ 次 FMA，$n_h - 1$ 个隐藏层间层中的每一层需要 H^2 次，输出层则需要 HN_W 次。因此，径向多层感知机每条边需要执行 $N_B H + (n_h - 1) H^2 + HN_W$ 次 FMA （ 7net-0 为 1,048 次），其中大部分发生在输出层，这是由于 N_W 较大。具体而言，仅输出层就需要 H 次 FMA（7net-0 中为 64 次），才能计算每个标量权重 $W_{p、u}(r)$。相比之下，式（4）中的 TP 只需少量 FMA 即可计算由每个 $W_{p、u}(r)$ 加权的乘积 $[y^{(l_1,u)}{}_j \otimes Y^{l_2}]^{l_3}$（例如，(1, 1, 1) 路径只需 6 次 FMA），因为克莱布施–戈尔丹系数大多因对称性而为零（补充信息图 S1b、c）。对 7net-0 一个交互层中的所有路径和通道求和后，TP 每条边仅执行 6848 次 FMA，约为径向多层感知机的十分之一；在其他模型中，径向多层感知机同样占主导地位（补充信息表 S1）。

在内存中，每个径向权重向量 W(r) 包含 N_W 个单精度浮点值，在每一层中每条边占用 $4N_W$ 字节（7net-0 为 3840 字节）。当径向多层感知机作为独立步骤进行计算时，前向传播必须将每条边的 W(r) 和多层感知机隐藏层激活值写入全局内存，而力的反向传播则必须在生成

同等大小的梯度 $\partial E/\partial \mathbf{W}$ 时将其读回。这些张量的总占用随边数和 N_W 成比例增长，因为它们按每条有向边存储，而非按原子存储；在凝聚相分子动力学中，每个原子通常在 r_c 范围内有数十个邻居。这种按边扩展的方式既会因内存流量拖慢执行速度，也会限制单个图形处理器可容纳的最大系统规模。对于 5632 原子电解质中的 7net-0（图 2a），仅径向权重 $\mathbf{W}$ 在五个相互作用层中就占用 3765MiB ，单层中最多可占用1036MiB 。

现有的张量积加速器会将径向多层感知机置于融合内核之外，并将 $\mathbf{W}(r)$ 作为从全局内存预先计算得到的输入，这主要出于两个实际考量。首先，将张量积与 $\mathbf{W}(r)$ 解耦，可使张量积内核不依赖于各模型的径向基和多层感知机架构，并将径向权重的计算交由标准稠密内核和自动微分处理。其次，将多层多层感知机直接融合到寄存器压力已很大的张量积内核中，需要在片上保留多层感知机的隐藏层激活值，并计算其力的反向传播，这很容易导致严重的寄存器溢出，同时每条边上繁重的多层感知机算术运算仍然无法消除。为了同时消除 $\mathbf{W}(r)$ 带来的内存流量和径向多层感知机的算术运算开销，JET-MLIPs 将融合卷积核与径向权重制表相结合（第 2.3 节）。

2.3 JET-MLIPs 卷积内核

与现有的张量积加速器类似，[26,28,29] JET-MLIPs 将邻居收集、张量积和散射融合到单个内核中。其内核遍历每个目标原子的关联边 $j \in \mathcal{N}(i)$ ， i 同时在寄存器中累积输出特征 h_i 。每个内核都针对层的维度进行编译；为满足寄存器预算的要求，必要时会将宽层划分为路径组（SI S2.3）。不同于现有的 TP 内核（后者从全局内存中读取预先计算的球谐函数 Y^{l_2} 和径向权重$\mathbf{W}(r)$（图 2a）），JET-MLIPs 将公式（2）–（5）中的完整逐边流水线融合到内核中（图 2b，SI 算法 S1）。当径向多层感知机直接在内核内部计算时（该内核仅模式已针对 SevenNet 的单精度浮点数和双精度浮点数实现，详见 SI

S3.1），前向传播会即时计算边距离 r、球谐函数 $Y^{l_2}(\hat{\mathbf{r}}_{ij})$、径向基 $B(r)$ 以及径向多层感知机，从而得到$\mathbf{W}(r)$，并在运行时直接累积寄存器中的张量积贡献，而无需将 $\mathbf{W}(r)$ 或任何其他逐边张量写入全局内存。

在力的反向传播过程中，现有的张量积加速器会将 $\partial E/\partial \mathbf{W}$ 和$\partial E/\partial Y^{l_2}$ 返回全局内存，以供外部反向传播步骤使用；而 JET-MLIPs 的融合反向传播内核会重新加载 y_j，重新计算 $Y^{l_2}(\hat{\mathbf{r}}_{ij})$，并将 $\mathbf{W}(r)$ 和 $\partial E/\partial \mathbf{W}$ 保留在寄存器中（SI 算法 S2）。在各路径组中，内核会累积 $\partial E/\partial y_j$，并在内核中应用球谐函数雅可比矩阵后，通过 GPU 原子加法计算边向量梯度 $\partial E/\partial \mathbf{r}_{ij}$。将 $\partial E/\partial \mathbf{W}$ 通过径向多层感知机（$N_B = 8$，SevenNet 中包含两个隐藏层，每层有 $H = 64$ 个单元）进行反向传播，有两种方法（SI S2.2 和 SI S4.7）。本文采用的分离内核方案（图 2b）中，第一个内核会将每条边对应的第二个隐藏层梯度 $\partial E/\partial \mathbf{a}_2$ 缓冲到全局内存中，第二个内核则完成 MLP 反向传播；另一种重计算方案则在每个路径组中重复执行 MLP 计算和反向传播，从而避免使用该缓冲区。尽管融合 MLP 可将 $\mathbf{W}(r)$ 和 $\partial E/\partial \mathbf{W}$ 从全局内存中移除，但两种方案仍需承担径向多层感知机的全部算术运算开销，并且在反向传播过程中必须缓冲或重新计算其隐藏层激活值。

为消除径向多层感知机运算以及仅内核模式的反向传播开销，JET-MLIPs 在其内核 + 表格模式（图 2c）中，用三次埃尔米特样条查表替代公式（2）和公式（3a、b）内核中的径向基和多层感知机。之所以能够进行这一替代，是因为训练好的模型的径向权重 $\mathbf{W}(r)$ 在公式（3a）中仅是边长 r 的固定函数，或在公式（3b）中对于每个元素对（Z_i、Z_j）是 r 的固定函数；这一方法借鉴了不变势[30,40] 中的径向制表以及 DPA4C 的协同设计架构[31]。对于每条边，融合前向内核直接从表中查找 $\mathbf{W}(r)$，而融合反向传播内核则利用同一个分段三次多项式计算 $\mathbf{W}(r)$ 及其径向导数 $\mathbf{W}'(r)$，从而得到径向梯度 g_r。在各路径组之间，

该内核将 g_r 以及相对于球谐函数的逐边梯度 g_Y，累加到一个缓冲区 (g_Y、 g_r) 中；该缓冲区包含 $(l_{\max}+1)^2+1$ 个每边数值，第二个内核通过球谐函数雅可比矩阵，将其对每条边转换一次为 $\partial E/\partial \mathbf{r}^{ij}$ 。该缓冲区是表格化卷积（图 2c）中唯一的逐边全局内存中间量。利用同一个多项式计算 $\mathbf{W}(r)$ 和$\mathbf{W}'(r)$ 还可确保计算力始终是表格化能量的精确负梯度，从而避免非保守力误差（第 2.4 节）导致的系统性能量漂移。

部署时，JET-MLIPs 通过在 N'间距为 $h = r_c/$（$N'-1$）的均匀网格节点上，以 fp64 计算 W(r) 和 W′(r)，覆盖 $[0, r_c]$，并将所得表以 fp32 存储；无论系统规模如何，每层在全局内存中仅占用 $N' \times 4N_W$ 个 fp32 值。对于具有元素对依赖的径向权重的模型（公式（3b），例如 MACE-MPA-0 和 MACE-OMAT-0），每层针对每个元素对构建一个表，并通过每条边的元素对索引进行选择；表的内存占用随体系中存在的元素对数量而扩展（补充信息 S6）。在内核中，为每条边和权重索引 c 加载四个 fp32 值，即可在前向传播中提供 W_c(r)的三次埃尔米特插值函数，并在反向传播中提供其径向导数 W_c'(r)；插值误差在 W_c(r)中按$\mathcal{O}$(h^4)扩展，并在 W_c'(r)中按 $\mathcal{O}$(h^3)扩展。[41] 为在第2.4节的精度预算内达到 fp32 舍入下限，表的构建过程先以 fp64 预计算 W_c 相邻节点之差，再转换为 fp32，以避免消去误差；在 SevenNet 和 EquFlash（$N' \le N$）中，使一个网格节点与截断起始半径 r_{on}对齐，从而让 f_{cut}(r)的二阶导数不连续性落在节点边界上；此外，将网格间距倒数 $1/h =$($N'-1$)$/r_c$表示为两个 fp32 数之和，以防止系统性的边长重缩放（补充信息 S2.4）。

2.4 精度预算

内核融合会计算模型的精确表达式，仅改变浮点舍入和求和顺序；而径向制表则用分段三次多项式近似径向多层感知机。我们根据未修改的数值本征

变化来评估这两项修改，并将这些基准变化定义为精度预算。当加速后端的偏差与现有 fp32 后端本身的偏差相当时，即视为满足该预算。在单次调用层面，主要尺度是：相同参数下，fp32 结果相对于 fp64 评估的舍入误差；其中 fp32 总能量还受 fp32 数值有限分辨率的限制。[42] 第二个静态尺度是对同一构型进行重复 fp32 调用时结果的离散程度，这是由 GPU 浮点加法不满足结合律所致，且会随线程求和顺序变化。[43,44] 仅以 fp64 评估径向表，还能进一步检验其单独的插值误差是否低于上述两种 fp32 尺度。在分子动力学轨迹中，混沌动力学会指数级放大单步差异，[45–47] 导致所有后端从相同初始状态出发的轨迹都会发生偏离。因此，轨迹级一致性通过恒能量（NVE）运行中的能量守恒[48] 以及热力学和结构观测量来评估；不同后端之间的差异必须与现有 fp32 后端重复运行所产生的差异相当。该预算是在经过测试的构型上采用的经验性接受尺度，而非统计等价性检验（补充信息 S3.1和 S3.2）。

2.5 模型适配器与分子动力学集成

要在无需重新训练的情况下运行第 2.3 节预训练模型中的内核，JET-MLIPs 为五个系列中的每个系列提供模型适配器（补充信息 S2.5）。对于给定的检查点，适配器会将每个交互层中的边嵌入、径向多层感知机、TP 和邻居聚合替换为融合卷积，同时保留其余模块不变，包括物种嵌入、线性投影和能量读出。转换后的模型可直接通过各系列的 ASE 计算器运行[35]，用于单点计算和弛豫工作流；也可通过 LAMMPS[34] 接口运行分子动力学模拟。在 LAMMPS 中，SevenNet 可通过其原生 `e3gnn` pair style 或机器学习原子间势（ML-IAP）接口运行，而其他四个系列则通过 ML-IAP 运行。为减少每步

分子动力学模拟期间的数据传输和内核启动开销，该集成仅在重建步骤将 LAMMPS 邻居列表上传至图形处理器，并在每个时间步长由图形处理器选择当前截断半径内的边。随后，它将模型评估作为单个 CUDA 图进行重放，把记录的 GPU 操作序列作为一个整体启动。还提供了一种 GPU pair style `jet/d3`（JET-D3），用于在 LAMMPS 中结合任意模型 pair style 计算 D3 色散校正（补充信息 S4.2），但除非另有说明，正文中的所有计时均不包含 D3。

2.6 计算细节

为在统一工作负载下比较内核和分子动力学集成，我们以一个包含 5632 个原子的液态电解质作为基准体系。该体系由碳酸乙烯酯（EC）和碳酸甲乙酯（EMC）中的 $LiPF_6$ 组成，晶胞构建和硬件规格见补充信息（SI）S1。立方模拟晶胞（$L = 38.74$埃）包含 64 个 $LiPF_6$，128 EC 和 256 个 EMC 化学式单元，对应 EC:EMC 质量比为 30:70，盐浓度为 1.83 mol L^{-1}，平衡密度为 1.361 g cm^{-3}。使用 7net-0 和 D3(BJ) 色散校正，通过 200 ps 的等温等压分子动力学对该晶胞进行了一次平衡，并将同一构型用于每个模型和后端（SI S1.1）。尽管该浓度高于典型锂离子电解质约 1 mol L^{-1} 的浓度，但此构型（SI 表 S2 中的原子数和边数）在此作为第 3.1 节精度评估和第 3.2 节 15 个模型性能比较的统一测试体系；而第 3.6 节的输运计算则使用以下所述组成的独立晶胞。在 15 个模型的比较中，我们根据预训练检查点的可训练参数数量，将模型分为小型（$< 10^6$）、中型（10^6–10^7）和大型（$> 10^7$）模型（SI 表 S2）。

在第 3 节中，“JET-MLIPs kernelonly”表示即时计算径向多层感知机的融合内核，而“JET-MLIPs kernel + table”表示使用径向表的融合内核（$N = 2048$，或 $N = 4096$ 对于 MACE）。在跨系列比较中，

（表 2）中，我们将 JET-MLIPs kernel + table 与每个系列未经加速的“上游模型”（SevenNet 使用 e3nn，MACE 和 NequIP-OAM 使用未转换的模型）以及其开发者发布的“现有 TP 加速后端”进行比较。具体而言，对于 SevenNet 和 MACE，该现有 TP 加速后端是通过 Kokkos ML-IAP 接口使用 cuEquivariance；对于 NequIP-OAM，则是通过 ML-IAP 使用 OpenEquivariance ；对于 EquFlash，则是结合 cuEquivariance 的 GGNN ML-IAP 封装。[49] 对于 Nequix，现有的 TP 加速后端指的是使用其软件包中的 OpenEquivariance 内核的 nequix PyTorch 模型；我们在 JET-MLIPs CUDA 图包装器中运行该模型，但未使用 JET-MLIPs 内核。在这些 LAMMPS 运行中，JET-MLIPs 通过 `e3gnn` 为 SevenNet（LAMMPS stable 2023 年 8 月 2 日更新 3）执行，并通过其 ML-IAP 封装器为其他系列执行（LAMMPS 2025 年 7 月 22 日更新 6）。除 EquFlash V2 外，所有模型均使用 CUDA 图重放。所有基准测试均在同一测量系列中，于同一节点的一块 NVIDIA H200 NVL GPU 上执行（硬件、软件版本和运行协议见 SI S1.2 和 SI S1.3）。

作为我们的主要计时指标，我们报告无重建步耗时（表 2 和补充信息图 S9c），其定义为第 50 步至第 500 步之间 45 个十步区间的每步耗时中位数。该时间窗口排除了启动阶段和 CUDA 图捕获；此外，取十步区间的中位数不会受到其中包含邻居列表重建的 9–10 个区间的显著影响（补充信息 S1.3）。包含重建的平均步耗时 $(t_{500}-t_{50})/450$（其中 t_n 表示第 n步的经过时间）则另行报告于补充信息 S4.1。除非图注另有说明，每个表项均为三次独立运行结果的中位数；而系统规模扫描中的每个数据点均采用其构型的第一次运行结果，并遵循在查看结果前预先确定的选择规则。

我们评估整个 LAMMPS 分子动力学运行期间的 GPU 内存消耗（全程峰值，表 2），以及在不同系统尺寸下、脱离 LAMMPS 独立进行能量和力评估时的 GPU 内存消耗（模型调用峰值，第 3.5节）。对于通过 Python 和 ML-IAP 执行的运行，我们记录 Py-

Torch（`torch.cuda.max_memory_allocated`）主动分配的峰值内存。对于通过 C ++ `e3gnn`pair style 执行的 SevenNet 运行，由于无法获取 PyTorch 内部的分配计数器，我们改为记录 `nvidia-smi`报告的 GPU 总内存。由于其中还包括 CUDA 运行时上下文以及内存分配器预留的缓存内存，因此该数值是模型内存使用量的上限（补充信息 S1.3）。

对于第 3.6 节中的输运应用，我们使用 7net-nano-6.0 和 D3(BJ)，通过 JET-MLIPs 内核 + table 和 LAMMPS 中的 JET-D3，计算了三种液体的剪切黏度。这三个体系分别为纯 EMC（670 个分子，10,050 个原子），以及两种 $LiPF_6$ 溶液：溶剂由 292 个碳酸乙烯酯分子和 440 个 EMC 分子组成，其中一种含有 65 个 $LiPF_6$ （10,040 个原子，0.94 mol L^{-1}），另一种含有 148 个 $LiPF_6$（10,704 个原子，2.02 mol L^{-1}）。每个晶胞均在恒压（1 bar）下于 298.15 开尔文平衡 1 ns，随后将体积重缩放至该次运行后半段的平均体积，并在恒定体积下再平衡 100皮秒。以此构型为起点，我们进行了三个恒定体积的 10 ns 生产模拟重复样本，各自采用独立的初始速度，并使用 Nosé–Hoover 恒温器将温度维持在 298.15 开尔文，时间步长为 1 飞秒。剪切黏度取自应力自相关函数的 Green–Kubo 积分随时间的变化所形成的平台；该结果对每 2 飞秒采样一次的五个独立剪切分量求平均，误差以三个重复样本的均值标准误表示（SI S5）。

我们还使用相同的模型、D3 校正和后端计算了硫银锗矿型 Li D PS_6Cl 的锂自扩散系数[5]。从有序结构的超晶胞（Materials Project mp-985592，11，*232* 个原子）开始，我们将一定比例 x的 Cl 原子与未和 P 成键的游离 S 原子进行交换。我们考虑了 $x = 0.25$、0.50 和 0.615，其中 $x = 0.615$ 与中子衍射测得的位点无序相匹配。[50] 每个晶胞先在 600 K 下退火 50皮秒 以重新分布 Li，随后在目标温度和 1 bar 下进行 200皮秒 平衡计算，并采用各向同性体积涨落；之后将晶胞固定在最后 100皮秒的平均体积。随后在恒定体积下，使用 Nosé–Hoover 恒温器和 2 飞秒 时间步长，在 300–400 K 下进行 4.5–20 ns 的生产运行。我们通过对 5184 个 Li 原子的均方位移（MSD）在滞后时间达到 D 14

MSD 达到 14埃^2 至运行时长一半之间的区间进行线性拟合，得到该值，并按照 He 等人的方法估算其相对标准差。[51] 随后根据 Nernst–Einstein 关系估算离子电导率 $\sigma = n_{\mathrm{Li}} e^2 D/(k_B T)$，并假设 Haven 比率为 1，其中 n_{Li} 为 Li 数密度，e 为元电荷，k_B 为玻尔兹曼常数，T 为温度（SI S5）。

3 结果与讨论

3.1 数值精度与分子动力学验证

我们首先在 5632 原子电解质晶胞的十个分子动力学帧上，评估了两种 JET-MLIPs 内核 + 表以及每个系列现有的 fp32 TP 加速后端相对于 fp64 参考值的单点能量和力误差（第 2.6 节和补充信息 S1.1）。其中，三个在 5632 个原子时进行 fp64 评估会超出图形处理器内存的模型（7net-omni、[52] NequIP-OAM L 和 EquFlash V1）在 352 原子晶胞上进行了测试。EquFlash V2 即使在缩减后的晶胞上进行 fp64 评估也会超出图形处理器内存，因此直接将其与现有的 fp32 TP 加速后端进行比较（补充信息表 S3）。所有 JET-MLIPs 内核 + 表评估均采用 N =4096 个 MACE 网格节点，并根据后续的网格收敛性分析，为其他四个系列采用 $N = 2048$ 个网格节点。在 SevenNet、NequIP-OAM 和 Nequix 中，两个后端的均方根（rms）能量误差均不超过 6.1×10^{-7} eV 原子$^{-1}$，约为每原子能量的 1×10^{-7}，且处于总能量 fp32 分辨率的两倍以内。最大差异是 NequIP-OAM L 中 JET-MLIPs 内核 + 表的常数偏移；在 fp64 下评估该表时，该偏移会消失，且不会改变力。补充信息以 fp32 总能量的最后一位单位（ulp）表示这些能量误差（补充信息 S3.1）。在 MACE 和 EquFlash 中，原子能量的 fp32 求和提高了基线能量误差，而 JET-MLIPs 内核 + 表的误差仍在现有 TP 加速后端能量误差的 1.01 倍以内。在具有 fp64 参考值的 14 个模型中，JET-MLIPs 内核 + 表的均方根力误差为现有 TP 加速后端的 0.54–1.02 倍，最大力分量误差为其 0.88–1.07 倍（表 1）。均方根比值

表 1：JET-MLIPs 内核力误差表 +，相对于现有的 TP 加速后端。对于每个具有 fp64 参考值的模型，JET-MLIPs 内核 + 表相对于 fp64 参考值的均方根力误差，最多为现有 fp32 TP 加速后端的 1.02 倍。相对于同一模型 fp64 评估值的力分量均方根误差以 10^{-6} 电子伏特埃$^{-1}$为单位，后面列出均方根误差和最大误差之比，即 JET-MLIPs 内核 + 表除以现有的 TP 加速后端（SevenNet 使用 FlashTP；十个电解质分子动力学帧；默认值 N；一块 H200）。SevenNet 的均方根误差比使用未舍入的误差，并给出跨帧的 95% 置信区间；其他比值采用 SI 表S3中的数值。因其 fp64 参考值超出图形处理器内存限制，故未列出 EquFlash V2。

模型	原子	现有的 JET-MLIPs		均方根比率	最大比率
7net-nano-5.5	5632	1.42	1.42	1.01 [1.00, 1.01]	0.89
7net-nano-6.0	5632	1.51	1.48	0.98 [0.98、0.99]	0.89
7net-0	5632	1.29	1.27	0.98 [0.98, 0.99]	1.03
7net-omni	352	1.27	1.27	0.99 [0.96、1.03]	0.88
MACE-MP-0 medium	5632	4.54	4.54	1.00	0.96
MACE-MPA-0 中型	5632	6.77	4.80	0.71	1.01
MACE-OMAT-0 中型	5632	8.96	4.86	0.54	0.95
MACE-MP-0 大型	5632	4.82	4.67	0.97	0.99
NequIP-OAM S	5632	7.70	5.33	0.69	0.99
NequIP-OAM M	5632	5.54	5.38	0.97	1.06
NequIP-OAM L	352	3.50	3.58	1.02	1.07
nequix-mp-1	5632	4.64	4.59	0.99	1.02
nequix-oam-1	5632	4.63	4.57	0.99	0.99
EquFlash V1	352	1.77	1.81	1.02	0.88

在其中 11 个模型中处于 0.97–1.02 范围内。其他三个模型（MACE-OMAT-0、NequIP-OAM S 和 MACE-MPA-0）的均方根力误差低于其现有的 TP 加速后端（比值分别为 0.54、0.69 和 0.71）。这一改进可能源于以 fp64 预计算径向权重，从而避免了径向多层感知机和距离变换中的 fp32 舍入（第 2.3节）。

JET-MLIPs 内核 + 表显示，在对同一构型进行重复的单精度浮点数评估时，其运行间变化幅度小于现有的 TP 加速后端（第 2.4 节）。在 15 个模型中，连续两次评估的差异在任何力分量上，对于 JET-MLIPs 内核 $\times\, 10^{-6}$ 埃的表$^{-1}$ 最多为 +，而现有的单精度浮点数 TP 加速后端的差异为 $(1.4\text{–}21) \times 10^{-6}\,\mathrm{eV\,Å}^{-1}$（补充信息表 S3）。JET-MLIPs 在每个模型中的变化范围都更小。JET-MLIPs 卷积的前向传播无需 GPU 原子操作

加法，因此具有确定性；这使得反向传播成为卷积中这种变化的唯一来源（补充信息 S3.1）。

径向表是 JET-MLIPs 中唯一的解析近似，其误差由网格节点数 N 决定。表的插值误差随径向权重网格间距的四次方以及其导数网格间距的三次方而减小（第 2.3 节和 SI S3.1），因此，将 N 增加一倍，会使径向权重的误差降低 16 倍，导数的误差降低 8 倍。测得的单精度浮点数力误差相应地随 N 增大而减小，直至达到现有后端的 fp32 舍入下限并趋于平台（图 3a、b 和 SI 图 S3）。除 MACE-MP-0 medium 在 $N = 4096$ 收敛外，所有受测模型均在 $N = 1024$–2048 时收敛；MACE-MP-0 medium 决定了 MACE 所用的较大表大小。为隔离表误差，我们选择了 SevenNet：其 ASE 接口以双精度浮点数形成边向量，因此在单点计算中具有最低的单精度浮点数舍入下限（SI S3.1）。对于其他系列，通过总单精度浮点数误差随 N 的收敛情况来检验该表（SI 图 S3）。在 SevenNet 中，当 $N = 2048$ 时，单精度浮点数 JET-MLIPs 内核 + 表与仅内核之间的最大力差为 1.0×10^{-5} eV Å^{-1}（图 3c），与单精度浮点数舍入下限相符。在双精度浮点数下隔离插值误差后，电解质中的均方根力误差比单精度浮点数舍入下限低 9–55 倍（SI 图 S2）；在钴酸锂$_2$, 、金刚石硅和非晶 SiO_2 中则低 43–188 倍（SI 表 S4）。电解质中制表造成的最大力分量扰动不超过1.8×10^{-6} eV Å^{-1}，远低于单精度浮点数 FlashTP 和 e3nn 的 (3.1–6.4) $\times 10^{-6}$ eV Å^{-1} 运行间差异。

在分子动力学中，混沌动力学会随时间呈指数级放大单步力差异。对于从相同初始位置和速度开始的 7net-nano-5.5 模拟，重复运行 FlashTP 时，两条轨迹之间的均方根原子位置差每 110 飞秒增大为原来的欧拉数倍；重复运行 JET-MLIPs 内核 + 表格时为 109 飞秒，而两个后端之间为 115 飞秒（补充信息 S3.2）。这些几乎相同的时间尺度表明，从现有的 TP 加速后端切换到 JET-MLIPs 内核 + 表格会导致

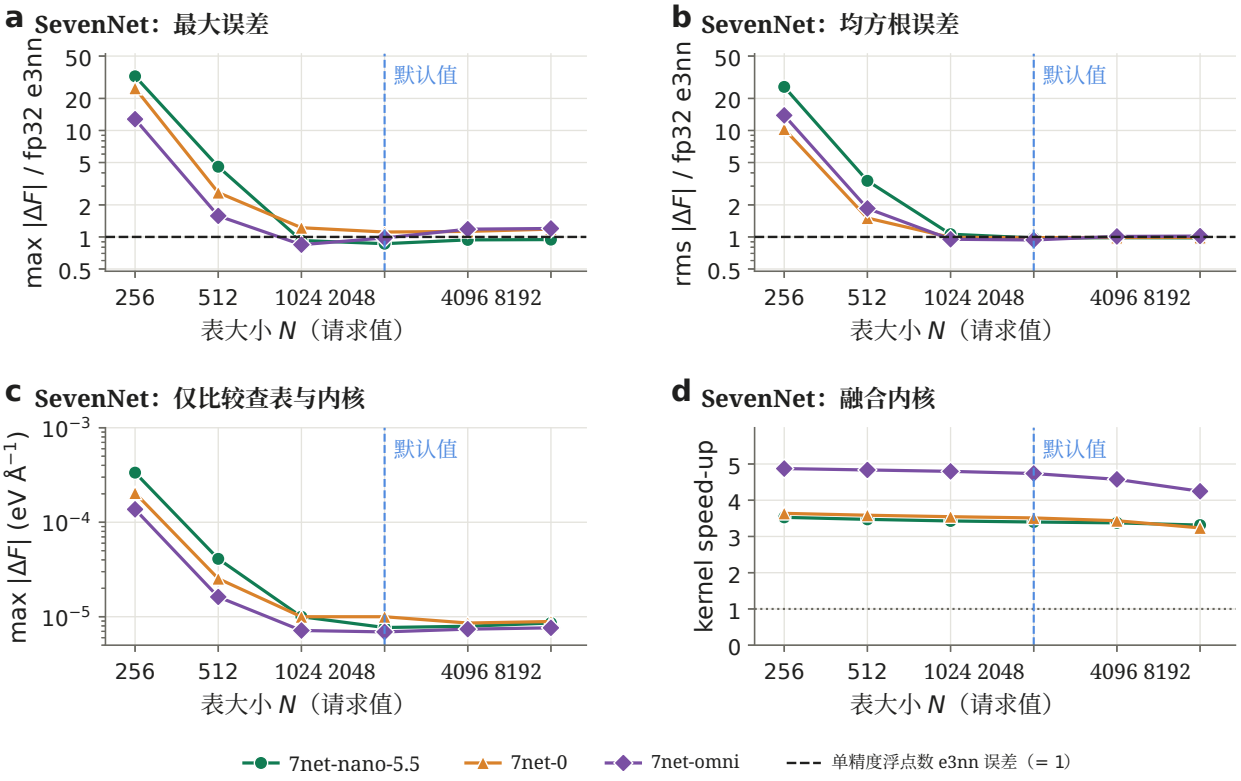


图 3：径向表精度和速度随网格大小的变化 N。(a、b) fp32 JET-MLIPs 内核 + 表相对于 fp64 e3nn 的最大力误差和均方根力误差，以 fp32 e3nn 的误差为基准进行归一化。(c) 与仅 JET-MLIPs 内核相比的最大力差（电子伏特 埃$^{-1}$）。(d) 按层汇总的前向传播加力反向传播内核加速比（内核耗时除以内核+ 表耗时；5632 原子电解质，单个 H200；(a、b) 中 7net-omni 使用 352 个原子）。竖直虚线标示默认值 $N = 2048$，水平线标示奇偶性。

轨迹的发散速度不超过使用相同后端重复模拟时的发散速度。表格化力仍是表格化能量的精确负梯度（见第 2.3 节），不会引入非保守力误差；微正则系综模拟也证实了能量守恒。在 3×10^{-5} 毫电子伏特/原子$^{-1}$ 皮秒$^{-1}$ 内，能量漂移幅度始终低于 100皮秒 的 7net-nano-5.5 和 7net-0 微正则系综模拟，分别使用 FlashTP、仅 JET-MLIPs 内核和 JET-MLIPs 内核 + 表。将 7net-nano-5.5 的模拟时长延长至 2纳秒，在两种 JET-MLIPs 模式下，漂移均未超过这一界限（补充信息 S3.2）。类似地，在 MACE-MP-0 medium 和 NequIP-OAM M 的三个随机种子、 + 表格以及现有的 TP 加速后端中，JET-MLIPs 内核的能量漂移范围与后者相重叠；这些模拟均为 10皮秒微正则系综模拟（补充信息 S3.2）。在 100皮秒 的 SevenNet 轨迹中，不同后端的热力学和结构观测量差异也与重复运行间的变化幅度相当，证实 JET-MLIPs 内核 +表支持稳定的分子动力学模拟（补充信息表 S5）。总体而言，JET-MLIPs 内核 + 表在所有 15 个模型的单点计算，以及在针对控制后端进行测试的四个模型的分子动力学模拟中，均满足第 2.4 节的标准；唯一例外是 NequIP-OAM L 存在可忽略的能量偏移。有关这些静态和动力学测试的完整详情，见补充信息 S3.1 和补充信息 S3.2。

3.2 15 个模型的步进时间

接下来，我们比较了 5632 原子电解质上全部 15 个模型的 LAMMPS 分子动力学步耗时，并与各模型系列现有的 TP 加速后端进行了对比（表2 和图 4a）。JET-MLIPs 内核 + 表格可加速每个模型：小型组的加速比为 3.0–5.7× ，中型组为 3.7–5.9× ，大型组为 1.5–4.8× 。在 0.5 飞秒时间步长下，就无需重建的吞吐量而言，这一加速使 5632 原子晶胞中超过1 纳秒/天$^{-1}$ 的模型数量从 3 个增至 12 个，其中 12.3 纳秒/天$^{-1}$ 对应 7net-nano-5.5 和 4.0，4.2 纳秒/天$^{-1}$ 则分别对应 7net-0 和 MACE-MP-0 medium。对于缺少原生 LAMMPS 接口的 Nequix，表2 将 JET-MLIPs 内核 +表格与已发布的 PyTorch 模型进行了比较；后者在

Table 2: LAMMPS step time and peak memory of the 15 models on one H200. JET-MLIPs kernel + table achieves the shortest step time and the lowest peak memory for every model. Speed-up divides the step time of the existing TP-accelerated backend by that of JET-MLIPs kernel + table, and the same-harness column reports this ratio when both backends execute inside the JET-MLIPs CUDA-graph ML-IAP wrapper (Sec. 3.3). Memory is the whole-run peak in GiB, where ≤ marks a whole-GPU upper bound (Sec. 2.6). Dashes denote unavailable or unmeasured entries, and OOM denotes out of memory. Step times are rebuild-free medians over five runs, or three runs for the upstream model and the same-harness comparison (5632-atom electrolyte, NVE, 0.5 fs, no D3, backends in Sec. 2.6). At this timestep, 43.2 ms step^{-1} corresponds to 1 ns day^{-1}.

	upstream model		existing TP-accelerated backend		JET-MLIPs kernel + table			same
model	ms step^{-1}	GiB	ms step^{-1}	GiB	ms step^{-1}	GiB	speed-up	harness
SevenNet, TP accelerator cuEquivariance								
7net-nano-5.5	112.8	–	19.9	2.67	3.5	≤1.04	5.7	3.2
7net-nano-6.0	143.8	–	21.1	3.39	4.1	≤1.06	5.2	3.4
7net-0	328.5	–	45.3	6.34	10.9	≤1.34	4.1	2.6
7net-omni	OOM	–	187.9	29.4	63.4	≤3.88	3.0	2.4
MACE, TP accelerator cuEquivariance								
MACE-MP-0 medium	318.1	33.4	60.3	18.4	10.3	0.76	5.9	4.1
MACE-MPA-0 medium	325.0	34.5	64.9	19.5	13.0	3.07	5.0	3.5
MACE-OMAT-0 medium	325.1	34.5	64.9	19.5	12.9	3.07	5.0	3.5
MACE-MP-0 large	395.3	30.5	64.2	14.6	13.4	1.49	4.8	3.1
NequIP-OAM, TP accelerator OpenEquivariance								
NequIP-OAM S	9.1	1.01	6.9	0.71	2.0	0.29	3.5	1.9
NequIP-OAM M	145.4	16.2	44.8	5.92	11.7	0.73	3.8	3.2
NequIP-OAM L	561.2	57.9	139.6	15.3	37.9	1.47	3.7	3.3
Nequix, PyTorch model with OpenEquivariance kernels in the CUDA-graph wrapper								
nequix-mp-1	–	–	74.0	11.8	25.0	1.21	3.0	3.0
nequix-oam-1	–	–	74.0	11.8	24.9	1.21	3.0	3.0
EquFlash, TP accelerator cuEquivariance in the GGNN ML-IAP wrapper								
EquFlash V1	–	–	868.7	114.7	365.2	10.7	2.4	–
EquFlash V2	–	–	1040.0	125.2	703.1	33.5	1.5	–

相同的 JET-MLIPs CUDA 图包装器。在每个参数规模组内，SevenNet 和 MACE 的加速比高于 NequIP-OAM、Nequix 和 EquFlash。参数数量衡量的是模型规模，而非计算成本；计算成本很大程度上取决于截断半径和邻居数量。例如，MACE-MP-0 large（15.8M 个参数，4.5 埃截断半径，每个原子有 35.5 条边）的步时间为 13.4 毫秒$^{-1}$，使用 JET-MLIPs 内核 + 表；而 NequIP-OAM L（9.6M 个参数，6.0 埃截断半径，每个原子有 88.2 条边）的步时间为 37.9 毫秒$^{-1}$。

在表 2中，SevenNet 通过原生 `e3gnn` pair style 与 JET-MLIPs 进行比较，而 cuEquivariance 则通过 ML-IAP 进行比较，因此这些比值包含了两种接口及其 LAMMPS 版本之间的差异（第 2.6 节）。双方都使用 ML-IAP 时，7net-nano-5.5、7net-nano-6.0、7net-0 和 7net-omni 相对于 cuEquivariance 的加速比分别为 5.1、4.7、3.8 和 2.9× （补充信息图 S9c）。在其他方面，表 2 中的 SevenNet 加速比是保守估计。首先，该表报告的是无重建步耗时，而 cuEquivariance 在邻居列表重建期间还会产生显著的额外开销（补充信息 S4.1）。其次，加入 D3 色散校正后，相对于 cuEquivariance 的加速比会进一步提高（补充信息 S4.2 及补充信息图 S9d）。

在 SevenNet 和 MACE 中，最宽的模型所实现的加速最小，分别为 3。0× 对于 7net-omni，以及 4。8× 对于 MACE-MP-0 large。这一趋势反映出，在更宽的架构中，径向制表和内核融合所加速的每一步计算占比更小。在 SevenNet 中，我们的运算次数（第 2.2 节）显示，径向多层感知机与张量积融合乘加运算的比值，从 7net-nano-5.5 中间层的 10.2 降至 7net-omni 最宽层的 4.1；而在首层和末层，由于参与门控输出的张量积路径较少，该比值达到 18–39（补充信息表 S1 和补充信息 S2.1）。因此，对径向多层感知机进行制表，在 7net-omni 中消除的每条边算术运算占比更小。在 MACE 中，图形处理器内核性能分析证实了类似的效应：MACE-MP-0 large 中融合卷积占每步图形处理器内核时间的 44%，而 MACE-MP-0 medium 中为 65%（补充信息 S4.3），因此每一步中有更大比例的运算位于卷积之外。

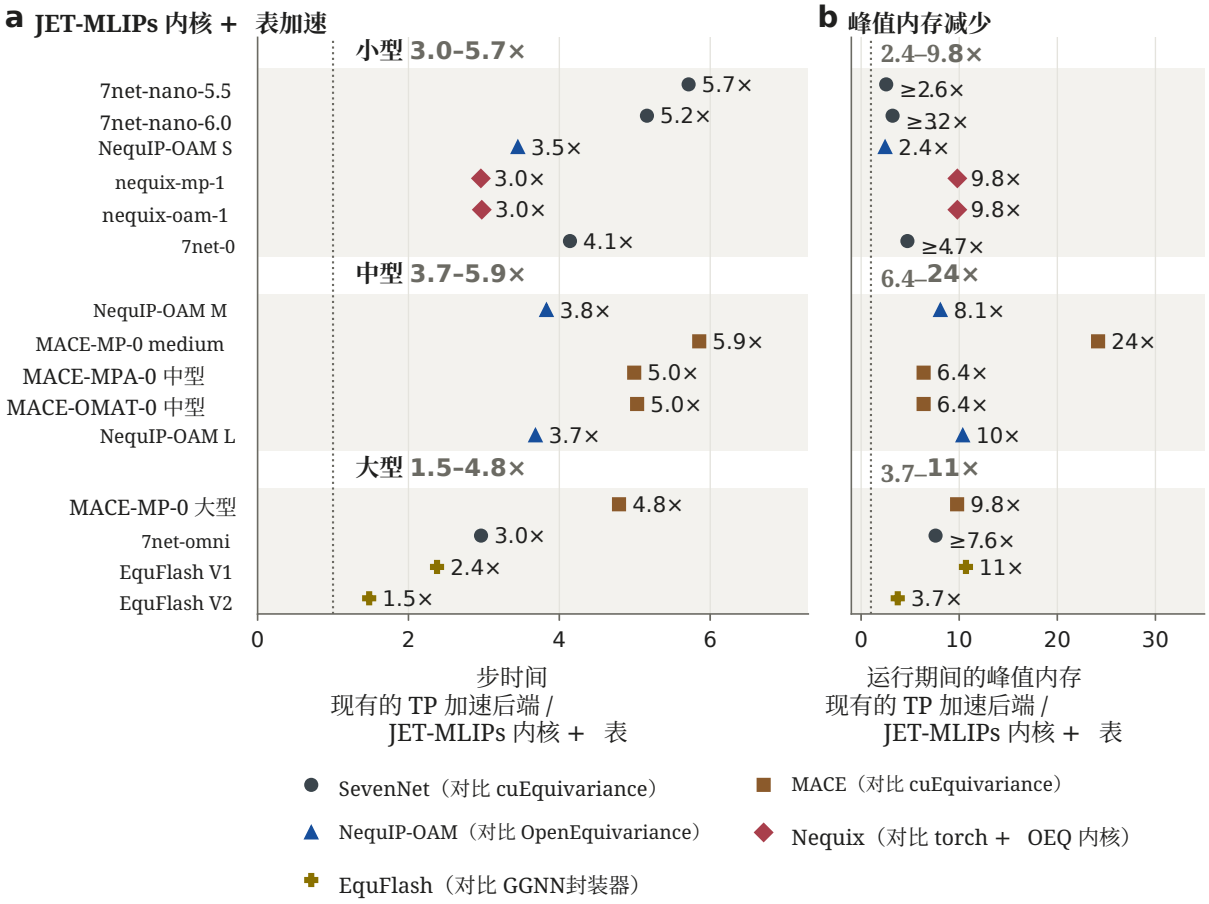


图 4：15 个模型的加速效果和内存减少量。(a) 无需重建的 LAMMPS 步进时间比，即现有的 TP 加速后端除以 JET-MLIPs 内核 +表。(b) 同理，全程峰值 GPU 内存比，根据未四舍五入的 PyTorch 峰值分配量计算（单次运行）。SevenNet 的比值代表下限，因为 JET-MLIPs 的内存使用量采用整块 GPU 峰值。在 (b) 中，MACE-MPA-0 和 MACE-OMAT- 0 的比值低于 MACE-MP-0 medium，因为它们为每个元素对存储径向表（SI S6）。两个面板均使用单张 H200 上的 5632 原子电解质，且不使用 D3。模型按参数数量排序，分为小型（$< 10^6$）、中型（10^6–10^7）和大型（$> 10^7$）组。填充颜色和标记用于区分系列，点线表示奇偶性。标签保留两位有效数字。GGNN 指 EquFlash 软件包。OEQ 指 OpenEquivariance。

3.3 加速分解

第 3.2 节的端到端加速比结合了径向制表、内核融合和 LAMMPS 集成，接下来我们将分解这些因素的贡献。SevenNet 是唯一提供仅内核模式的系列，因此可以直接隔离径向制表的影响。将内核内的径向基和多层感知机替换为三次 Hermite 表，可使融合卷积块的耗时减少约 70%，在 7net-nano-5.5 中从 7.5 ms 降至 2.3 ms，在 7net-0 中则从 27.3 ms 降至 8.0ms（SI 图 S9a）。在针对 5632 个原子进行的独立计时中，采用三次 Hermite 表的逐层求和融合卷积内核，运行速度比仅内核模式快 3.4–4。7× 倍（图 3d），并将整个模型调用的耗时缩短 1.7–3。2×倍（SI S4.4）。由于路径组数量的缘故，这两项提升在 7net-omni 上最大。消除内核内多层感知机隐藏状态后，采用制表的内核只需在寄存器中存储插值权重、球谐函数和张量积累加器，使每个路径组能够在寄存器预算内容纳更多路径和通道。因此，在 7net-omni 最宽的层中，采用制表的反向传播只需 8 个路径组，而仅内核模式则需要 32 个路径组；后者的每个路径组都必须重新遍历关联边并重新计算多层感知机（SI S2.3）。相比之下，现有张量积加速器并未减少径向多层感知机运算，这与第 2.2 节的运算次数相符（SI S4.3）。

不进行径向制表时，加速幅度会显著降低，并且高度依赖 LAMMPS 集成。通过 SevenNet 原生的 `e3gnn` pair style，JET-MLIPs 内核仅比 cuEquivariance 快 2.3、2.0 和 1。5× 倍，分别对应 7net-nano-5.5、7net-nano-6.0 和 7net- 0，但对于 7net-omni 仅达到其 0。9× 倍的速度（补充信息图 S9c）。然而，在 LAMMPS 之外进行独立模型调用时，对于 7net-0 和 7net-omni，cuEquivariance 的速度快于仅内核模式，这表明仅内核模式在 LAMMPS 中的端到端优势源于集成，而非单纯来自卷积。在 GPU 上进行邻居列表筛选和边构建，是这一集成加速收益中占比最大的一部分（补充信息 S4.5 和补充信息图 S9g）。对于仅内核模式落后于 cuEquivariance 的 7net-omni，引入径向表后，加速恢复并提升至 3。0×倍。

为了区分不同系列中的内核加速收益与部署开销，我们将每个预训练模型都置于相同的 JET-MLIPs CUDA 图包装器中运行，使唯一不同之处在于卷积后端（表 2的相同测试框架列）。在这一控制条件下，与其现有的 TP 加速后端相比，JET-MLIPs 内核 +表明其在 SevenNet、MACE 和 NequIP-OAM 中快 1.9–4。1× 倍，而端到端加速为 3.0–5。9× 倍（Nequix 则在相同包装器中评估，达到 3。0×倍）。对于 NequIP-OAM M，3。8× 倍的端到端加速因此可分解为：包装器带来的 1。2× 倍和内核带来的 3。2× 倍（补充信息 S4.5）。在 MACE 中，现有的 TP 加速 ML-IAP 部署通过 Python 逐个操作急切地执行模型，而 JET-MLIPs 包装器则使用 `torch.compile` 编译非卷积块，并将这一步作为 CUDA 图重放（补充信息 S2.5）。因此，表 2 中的每个端到端加速都反映了内核因子（在相同测试框架列中报告）与部署因子的乘积。随着卷积时间降低，剩余运行时间中有相当一部分转移到卷积之外的操作上（补充信息 S4.3 和补充信息图 S6），这与压缩通用机器学习原子间势中报告的固定分子动力学步成本相呼应[31]，并表明非卷积模块构成了下一项性能瓶颈。

3.4 跨材料的系统规模缩放

为评估超越 5632 原子晶胞后的尺度扩展，我们针对九个模型，在电解质副本和三种致密固体上扫描系统规模，并与其现有的 TP 加速后端进行比较，即钴酸锂、$_2$, 金刚石硅和非晶硅$_2$（图 5a–i 及补充信息 S4.6）。在每个现有 TP 加速后端所能容纳的最大晶胞范围内，加速效果一直保持存在，始终不低于 1。4× ，但其随系统规模的扩展趋势因模型而异。在全部四个 SevenNet 模型中，与 cuEquivariance 相比的加速效果在每种材料中都会随规模增大而降低（图 5a–d）。在电解质副本中，对于 5632 个原子，其加速比分别从 5.8、5.2、4.1 和 3.0×下降（其中 5.8× 反映的是首次运行扫描约定，而不是表 2、补充信息 S1.3中的五次运行中位数）至 3.7、3.9、3.0 和 2.6×，分别对应 cuEquivariance 所能容纳的最大规模（两个纳米模型为 152,064 个原子，7net-0 为 45,056 个原子

以及 22,528 个原子，对应 7net-omni）。在三种致密固体中，下降趋势更为陡峭；这些固体的最小晶胞包含 70–360 个原子，7net-nano-5.5 的加速比从 11.0–14.0× 降至最大规模下的 3.4–4.2×（这些固体中的静态精度已在第 3.1节得到验证）。在其他系列中（图 5e–i ），NequIP-OAM M 以及 EquFlash V1 和 V2 （在最小晶胞以上）也表现出类似的随规模增大而下降的趋势。相比之下，MACE-MP-0 medium 仅在达到几千个原子之前下降，随后在 5.4–6.1×处趋于平台；而 nequix-mp-1 则随规模增大而上升，在最大晶胞下达到 2.9–3.4× 。

7net-nano-5.5 的步时分解表明，随体系规模增大，加速比下降源于现有 cuEquivariance 后端的固定每步开销，而不是 JET-MLIPs 效率降低（SI S4.5）。具体而言，即时执行模型和机器学习原子间势接口的主机端启动会产生与原子数量无关的下限，使两个纳米模型在 14ms 左右的 cuEquivariance 步时间几乎保持不变，适用于低于 2000 个原子的钴酸锂$_2$ 晶胞。随着体系增大，这一固定成本会分摊到更多原子上，加速比也会趋近于两个后端单原子成本之比。预计其他 SevenNet 检查点和 NequIP-OAM M 也会呈现相同机制，因为它们现有的 TP 加速后端使用相同的接口；而 MACE-MP-0 medium 在几千个原子以上观察到的平台现象仍未得到解释。总体而言，小晶胞的加速比主要反映每步启动开销，而大晶胞的加速比则反映内核吞吐量。

步时间相对于原子数的幂律拟合（$t \propto N^{\alpha}$ 对于 $N \geq 4000$ 个原子）定量区分了这三种缩放机制（补充信息图 S8）。在全部九个模型中，JET-MLIPs 内核 + 表呈现近线性尺度扩展，指数为 $\alpha = 0.87$–0.98。在现有的 TP 加速后端能够处理的尺寸范围内，其指数 α 对 SevenNet 和 NequIP- OAM M 而言较小（0.68–0.89），对 MACE-MP-0 medium 而言相近，对 nequix-mp-1 而言较大，这与加速分别递减、持平和上升的趋势直接吻合。在超过各自上限的下一个扫描尺寸下，cuEquivariance 在 LAMMPS 中运行全部四种 SevenNet 模型时均出现内存不足，而 JET-MLIPs 内核 + 表仍可继续运行（第 3.5 节及补充信息 S4.7），从而改变

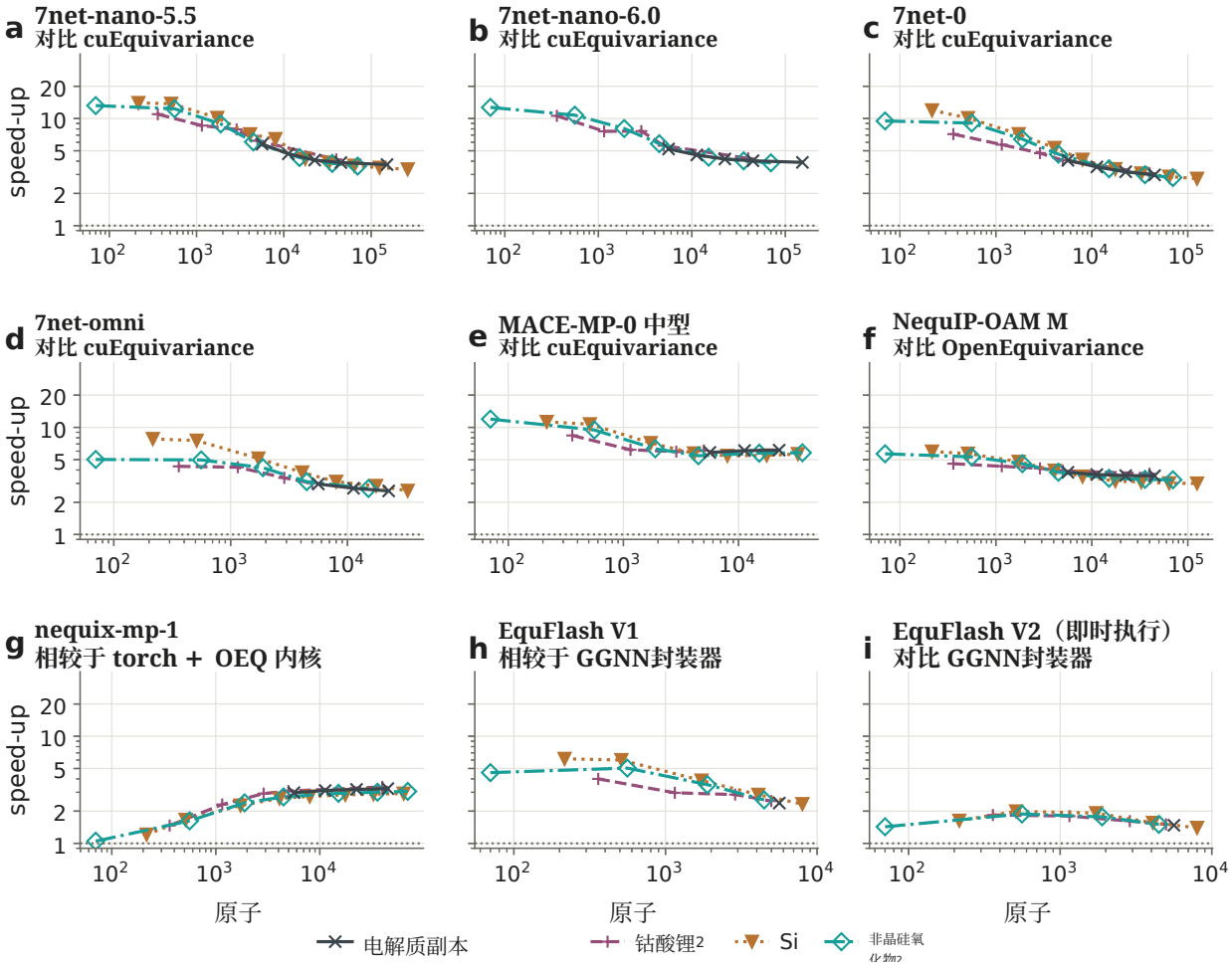


图5：不同材料中加速比随系统规模的变化。(a–d) SevenNet 的现有 TP 加速后端步时间与 JET-MLIPs 内核 + 表时间之比。(e–i) MACE、NequIP-OAM、Nequix 和 EquFlash 的相同比值，各面板中标注了模型和比较对象（默认值表格、单张 H200、无需重建的 LAMMPS 步骤、每个数据点仅运行首次）。体系分别用叉号（电解质）、加号（钴酸锂$_2$）、向下三角形（硅）和空心青色菱形（非晶硅氧化物$_2$）表示。除 SevenNet 硅和 SiO_2, 采用 460 个无需重建的步骤外，所有运行均使用 500 步。EquFlash V2 运行时不使用图重放。仅对 SevenNet 验证了这些固体的静态精度，且 7net-nano-6.0 未进行 cuEquivariance硅基准测试。曲线在各现有 TP 加速后端所能处理的最大实测系统规模处结束。OEQ 指 OpenEquivariance。

从速度到可行性，对大型规模进行比较。

3.5 GPU 内存尺度扩展与可实现的系统规模

GPU 内存决定了设备可模拟的最大体系规模，也决定了哪些硬件能够运行给定模型。因此，我们接下来评估了 15 个模型的 LAMMPS 基准测试（第 3.2 节）全程峰值内存消耗，以及独立模型调用的峰值分配。与现有的 TP 加速后端相比，全程峰值在小型组中降低了 2.4–9。$8\times$ ，在中型组中降低了 6.4–$24\times$ ，在大型组中降低了 3.7–10。$7\times$ （表 2 和图 4b）。对于 SevenNet，JET-MLIPs 通过 `e3gnn` 运行并报告所用 GPU 内存总量（第 2.6 节），相较于 cuEquivariance 的内存降幅下限分别为：7net-nano-5.5 至少 2。$6\times$ ，7net-nano-6.0 至少 3。$2\times$ ，7net-0 至少 4。$7\times$ ，7net-omni 至少 7。$6\times$ 。在所有三个组中，JET-MLIPs 内核 + 表明，每个模型的 GPU 内存消耗和步时间均有所降低。

为在不同系统规模下独立于 LAMMPS 分析模型调用的内存分配，我们测量了九种模型单次能量和力评估的 PyTorch 峰值内存（图 6）。制表使内存分配量低于所有上游后端和现有的 TP 加速后端，其中 SevenNet 的内存分配量降低了一个数量级。在 5632 个原子时，制表后的 7net-nano-5.5、7net-0 和 7net-omni 调用分别需要 0.21、0.42 和 1.92 吉比字节，分别比 FlashTP（2.22、5.48 和 26.1 吉比字节）少 11–$14\times$ 倍，比 cuEquivariance（2.16、5.35 和 25.3 吉比字节，图 6a、c、d）少 11–$13\times$ 倍。同样，7net-nano-6.0 需要 0.21 吉比字节，而 cuEquivariance 需要 2.75 吉比字节（图 6b）。在 5632 个原子的其他系列中（图 6e–i），JET-MLIPs 内核 + 制表对 MACE-MP-0 medium、NequIP-OAM M 和 nequix-mp-1 分别使用 0.93、1.08 和 1.04 吉比字节；相较各自现有的 TP 加速后端，内存分别减少 7.4、6.8 和 10.$2\times$ 倍，而 EquFlash V1 和 V2 使用 10.5 和 32.0 吉比字节（比 GGNN 封装器少 10.7 和 3.$8\times$ 倍）。在 SevenNet 系列中，JET-MLIPs 仅内核的重计算方案所用内存甚至低于内核 +制表方案，因为它不存储逐边缓冲区，但代价是需要重复进行径向MLP评估

(SI S4.7 和 SI 表 S6）。

这些内存节省直接转化为可处理的系统规模大幅提升。在单个 H200 上进行独立模型调用扫描时，JET-MLIPs 内核 + 制表可处理 7net-nano-5.5 和 7net-nano-6.0 的 2,883,584 个原子、7net-0 的 1,441,792 个原子，以及 7net-omni 的 360,448 个原子。cuEquivariance 可处理两种纳米模型的 180,224 个原子、7net-0 的 90,112 个原子，以及 7net-omni 的 22,528 个原子。对于 MACE-MP-0 medium、NequIP-OAM M、nequix-mp-1、EquFlash V1 和 EquFlash V2，可处理的原子数分别为 720,896、360,448、720,896、45,056 和 22,528；作为对比，MACE-MP-0 medium 和 NequIP-OAM M 的现有 TP 加速后端可处理 90,112 个原子，现有 Nequix 后端可处理 45,056 个原子，而 GGNN 封装器可处理 5632 个原子。这些数值表示已验证的规模，而非精确的容量上限，因为扫描过程中相邻测试点的原子数会加倍（模型调用计时和 SevenNet 其他后端的数据见 SI S4.4 和 SI 图 S7）。在针对 MACE、NequIP-OAM、Nequix 和 EquFlash V1 进行的完整 LAMMPS 规模扫描中（SI 表 S7），在 JET-MLIPs 内核 + 制表仅占 GPU 总内存 4–55% 的系统规模下，现有 TP 加速后端会出现内存不足，或占用 GPU 总内存的 80–99%。

较低的内存占用也使这些工作负载能够在消费级 GPU 上运行。在适用于 Linux 2 的 Windows 子系统（WSL2）下，使用 12 GB RTX 4070 SUPER（SI S4.8），JET-MLIPs 内核 + 表明，该体系在 5632 原子电解质上的运行速率分别为 6.27、1.54 和 0.185 纳秒/天$^{-1}$，对应于 7net-nano-5.5、7net-0 和 7net-omni。相比之下，FlashTP 和 cuEquivariance 均无法容纳 7net-omni，且只有 FlashTP 能容纳 7net-0 （ SI 图 S9e ）。在这些运行所采用的 0.5 飞秒时间步长下，消费级 GPU 的速率（根据包含重建操作的平均步耗时计算）超过了现有 TP 加速后端在相同指标下所达到的最高 H200 速率，即使用 OpenEquivariance 时三个模型分别为 2.24、0.86 和 0.132 纳秒/天$^{-1}$。在同一块 12 GB GPU 和 5632 原子晶胞上（SI 图 S9f），JET-MLIPs 内核 + 表明，MACE-MP-0 中型、MACE-MPA-0 中型和 MACE-MP-0 大型模型的运行速度分别为 28.4、41.2 和41.9 毫秒/步$^{-1}$，而 cuEquivariance 对这三个模型均无法运行。

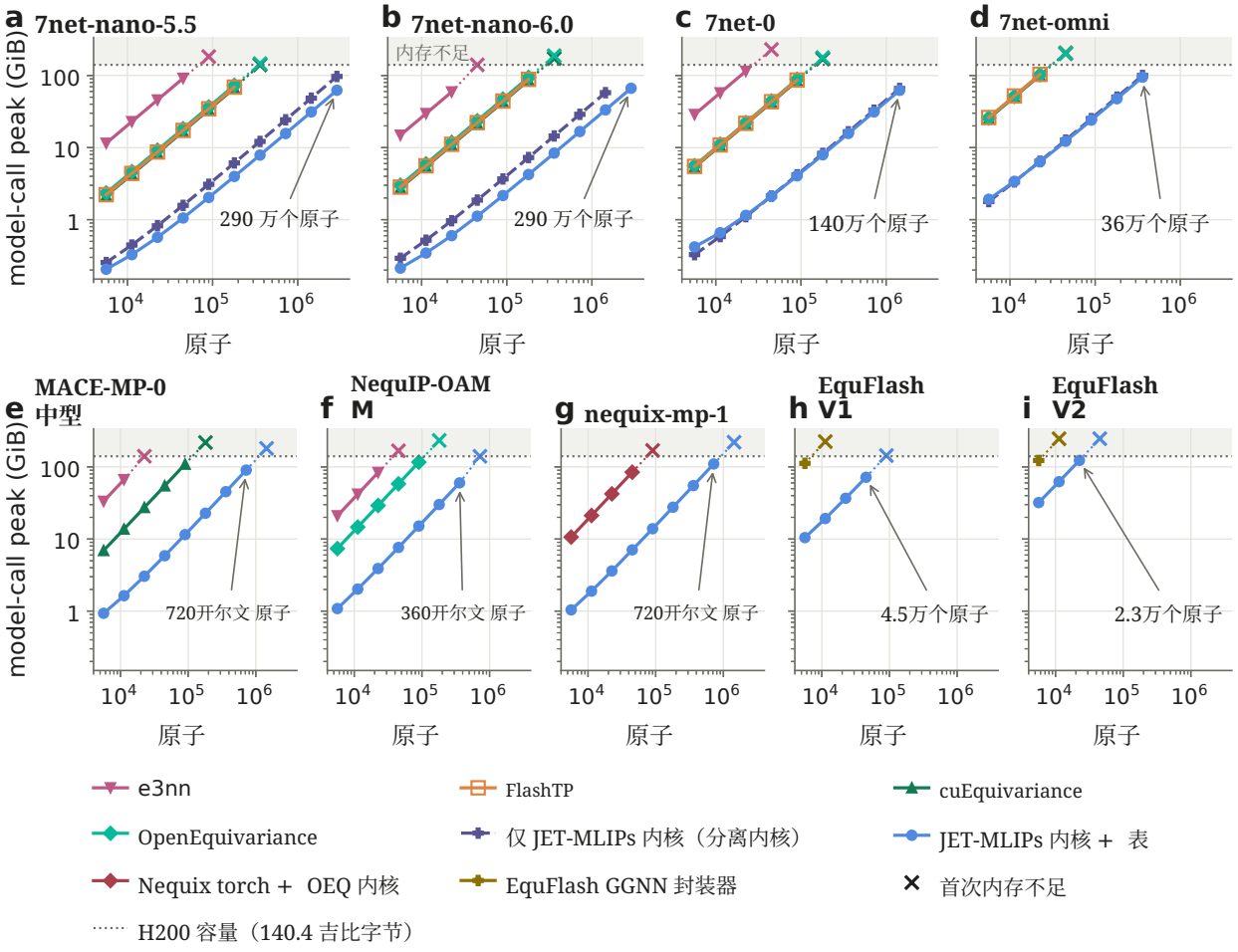


图 6：模型调用内存与可处理的系统规模。 在原子数（5632 原子电解质副本、一个 H200，前向计算加力计算）下，以吉比字节为单位的 PyTorch 峰值分配。（a–d）SevenNet 模型。（e–i）MACE、NequIP-OAM、Nequix 和 EquFlash 模型，各模型在对应面板中标注；使用原生接口且不进行图重放。仅 JET-MLIPs 内核使用分离内核方案（虚线），而 JET-MLIPs 内核 + 表对 $N = 4096$ MACE 使用，对其他模型使用 2048。后端标记的定义见图例。十字标记表示首次内存不足的系统规模及其真实原子数，并通过点状线性外推连接至可容纳的最大规模；若该外推线低于内存容量线，则在内存容量线处截断。灰色带表示内存不足，水平点线表示 141 GB H200 的 140.4 吉比字节容量，面板注释则表示 JET-MLIPs 内核+ 表可容纳的最大系统规模。SevenNet 的 TP 加速器曲线彼此重叠在 10% 以内，(c,d) 中的 JET-MLIPs 曲线也是如此。在 (d) 中，e3nn 在 5632 个原子时超过图形处理器内存，因此没有曲线或标记；在 (h,i) 中，GGNN 只能容纳 5632 个原子。OEQ 表示 OpenEquivariance。

3.6 长时间尺度输运性质的应用

为展示这项加速如何使长时间尺度模拟成为常规操作，我们将 JET-MLIPs 内核 + 表与 JET-D3 结合，用于研究两种需要对约 10^4 个原子进行纳秒级轨迹模拟的输运性质（计算细节见第 2.6 节，结果见表 3 和表 4）。第一种性质是三种碳酸酯液体的格林–久保剪切黏度；每种液体均通过三个独立的 10 纳秒 NVT 副本进行采样，生产分子动力学模拟总时长为 90 纳秒。根据在 0.94 mol L^{-1} 电解质晶胞上使用 D3 测得的 LAMMPS 步耗时（补充信息图 S9d），采用上游 e3nn 在一块 H200 上运行单种液体的三个 10 纳秒黏度副本需要 129 天，使用 FlashTP 则需要 76 天，而采用 JET-MLIPs 内核+ 表与 JET-D3 只需 3.9 天。对于三种液体，按相同步耗时计算，使用上游 e3nn 需要一年多的专用 H200 计算时间，使用 FlashTP 则需要七个多月，而采用 JET-MLIPs 内核 + 表与 JET-D3 只需 12 天。在 2.02 mol L^{-1} 溶液的生产模拟中，三个副本共用一块 H200，每个副本的推进速度为每天 2.87 纳秒$^{-1}$，因此完成三个 10 纳秒副本需要 3.5 天。相应的加速比与上述讨论的不同，因为这些步耗时还包括由 JET-D3 加速的 D3 色散校正（补充信息 S4.2）。

298.15 开尔文时，液态电解质的 Green–Kubo 剪切黏度分别为：纯 EMC 为 0.956(14) mPa s，0.94 mol L^{-1} $LiPF_6$ 溶液为 3.17(15) mPa s，2.02 mol L^{-1} 溶液为 9.27(33) mPa s。Green–Kubo 积分随时间的变化在纯 EMC 中于 30–75 皮秒达到平台，在 0.94 mol L^{-1} 溶液中于 50–100 皮秒达到平台，这表明应力相关时间与黏度一样，会随盐浓度增加而延长（SI S5）。作为比较，据报道，EC/EMC（质量比为 3:7）中 1 mol L^{-1} $LiPF_6$ 溶液在 25 °C 下的剪切黏度为 3.2 mPa s。[53]

在硫银锗矿型 Li_6PS_5Cl 中，温度为 400 开尔文 时的锂自扩散系数为 9.17×10^{-7}，7.54×10^{-7} 和 $7.09\times10^{-7}\,\mathrm{cm^2 s^{-1}}$，对应于 $x=0.25$、0.50 和 0.615，三种组成的相对标准偏差为 5–7%（表 4）。在 350 至 400 开尔文 之间进行阿伦尼乌斯拟合，得到三种组成的活化能分别为 0.24 ± 0.02、0.28 ± 0.02 和 $0.29(2)\mathrm{eV}$。该数值的下降

D 从 $x = 0.25$ 到 0.615，与先前一项基于机器学习间势的分子动力学研究结果一致；该研究发现电导率最大值接近 $x = 0.25$，[54] 而其他模拟则将最大值置于接近$x = 0.5$的位置。[55,56] 在 300 开尔文和 $x = 0.615$下，20 纳秒轨迹得到 $D = 4.33 \times 10^{-8}\,\mathrm{cm^2\,s^{-1}}$（相对标准偏差为 6.7%），对应的 Nernst–Einstein 离子电导率为 6.95 mS $\mathrm{cm^{-1}}$。该数值高于对 x 在 0.52 至 0.62 之间的样品测得的 2–5 mS $\mathrm{cm^{-1}}$。[50,54,57,58] 在所有温度下，拟合窗口内的均方位移均呈轻微次线性变化（对数–对数斜率为 0.80–0.88），这主要源于笼效应贡献；将拟合限制在窗口的后半段后，D 降低约 1%，完全处于统计不确定性范围内。作为传输层面的一致性检查，我们使用 FlashTP 重复了$x = 0.615$ 在 400 开尔文下的计算，以 FlashTP 替代 JET-MLIPs 内核 + 表，同时保持模型、D3 实现、初始结构和分析方法不变。两个后端得到的结果分别为 $D = 7.07 \times 10^{-7}$和$7.09 \times 10^{-7}\,\mathrm{cm^2\,s^{-1}}$，相差 0.2%，而相对标准偏差为 5.5%；两者的平衡体积一致到 0.01% 以内。

表 3：使用 JET-MLIPs 内核计算的碳酸酯液体剪切黏度+ 表和 JET-D3。 剪切黏度 η 在 298.15 开尔文 三个重复样本的标准误差（采用 D3(BJ) 的 7net-nano-6.0，计算细节见第 2.6节）。

液体	原子	η (毫帕·秒)
EMC	10,050	0.956 ± 0.014
0.94 mol $\mathrm{L^{-1}}$ $\mathrm{LiPF_6}$ （EC/EMC 中）	10,040	3.17 ± 0.15
2.02 摩尔/升$^{-1}$ $\mathrm{LiPF_6}$ 在 EC/EMC 中	10,704	9.27 ± 0.33

表 4：使用 JET-MLIPs 内核 $_6$PS$_5$表和 JET-D3 计算的 Li + 自扩散。 锂自扩散系数 D 在 Cl/S 位点交换分数 x 下的活化能 E_a 由 350 至 400 开尔文 范围内的 Arrhenius 拟合得出（使用 D3(BJ) 的 7net-nano-6.0，计算细节见第 2.6节）。

	D (10^{-7}厘米^{2}秒$^{-1}$)			
x	350 开尔文	375 开尔文	400 开尔文	E_a (电子伏特)
0.25	3.46	5.88	9.17	0.24 ± 0.02
0.50	2.39	4.54	7.54	0.28 ± 0.02
0.615	2.13	4.17	7.09	0.29 ± 0.02

3.7 可迁移性与局限

我们的 fp32 精度结果（第 3.1 节）似乎与 DP Compress 的结论相矛盾。DP Compress 发现，其压缩模型在 fp32 下的性质误差并未随表间距缩小而降低，因此建议使用 fp64。[30] 误差若与表间距无关，就表明制表并非其主要来源；我们的径向表也呈现出同样的平台，在达到 fp32 舍入下限 $N = 1024$–2048（图 3a、b）后便如此。差异在于评估标准，而非算术运算。我们的精度预算以 fp64 参考值为基准，确定未修改的现有后端的 fp32 舍入下限，并在 fp64 下单独评估表误差；在电解质中，该误差比 fp32 舍入误差低 9–55 倍，在三种致密固体中则低 43–188 倍。相比之下，DP Compress 在 fp32 测试中既未报告未压缩模型的 fp32 误差，也未报告参考值的精度。因此，其 fp32 误差上限似乎反映的是 fp32 舍入误差在派生性质中的放大，而非制表本身；相较之下，以现有 fp32 后端设定的精度预算为基准时，插值误差远低于固有单精度浮点舍入误差。

除了已测试的五个系列，只要径向权重仅取决于原子间距离和元素对，径向制表就适用。在 Matbench Discovery 的领先模型中，[21] GRACE[59] 和 TACE[60]的径向 MLP 直接满足这一条件。在 EquiformerV3 中，[61]eSEN[48]和 TECE[62]只有径向函数可以制表，因为它们的注意力或混合操作取决于节点特征。径向输入直接包含节点特征的模型（如 Prophet）[63] 则不适用此方法。这一条件也适用于原子间势之外的领域。TrajCast 是一种自回归等变网络，它直接预测原子位移和速度，而非力；它通过 NequIP型卷积构建消息，其径向权重仅取决于边长。[64] 虽然我们尚未测试 TrajCast，但径向制表和融合前向内核应可沿用于其消息传递步骤；而其依赖速度的更新不属于当前内核的适用范围，也无需进行力的反向传播。即使径向制表适用，共享等变性本身也无法保证与其他架构兼容——

例如 EquiformerV2 的高阶表示。[65]因此，在复用这些内核之前，必须针对支持的布局（SI S6）和架构相关的编译限制（SI S2.3）验证每种新架构。

在所测试的系列中，我们从三个层面验证了各模型的精度。针对 SevenNet，我们依据舍入误差、不同运行间的差异以及混沌放大的完整精度预算进行测试，并以 FlashTP 作为轨迹发散的对照，以 JET-MLIPs 仅内核作为表格的对照。这些动力学测试涵盖了在一块 H200 上运行的 5632 原子电解质晶胞（第 3.1 节）；单点力和能量误差则将验证范围扩展至钴酸锂$_2$, 金刚石硅和非晶 SiO_2 (补充信息表 S4)。我们通过与 fp64 参考值比较单点误差，并与现有的 TP 加速后端比较 NVE 漂移，验证了 MACE-MP-0 medium 和 NequIP-OAM M。其余 MACE 和 NequIP-OAM 模型、Nequix 以及 EquFlash V1 均通过与 fp64 参考值比较单点误差进行测试；对于 EquFlash V2（其 fp64 模型超出 GPU 内存容量），则直接与其现有的 fp32 TP 加速后端比较。我们还检查了 Nequix 和 EquFlash V1 在仅使用 JET-MLIPs、未运行现有 TP 加速后端对照组时的 NVE 漂移（补充信息 S3.2）；分子动力学轨迹发散仅针对 SevenNet 进行了测量，而其他四个系列则通过重复单点计算进行测试（补充信息表 S3）。必须针对每个模型分别确定精度预算，因为该预算取决于各模型自身现有的 fp32 后端（第 3.1 节）；因此，在致密固体中进行动力学验证，以及针对更多电解质组成开展精度检查，是最直接的后续扩展方向。

性能基准测试存在若干范围限制：计时仅涵盖一种电解质组成和三种固体，且必须结合其邻居数进行解读。多GPU执行不在本研究范围内（补充信息 S6）；逐边类型截断、NequIP-OAM XL 和 7net-l3i5层配置目前仍不受支持，Al- legro 的边特征张量积也尚未评估。不同工作负载的加速效果各异，不存在普遍适用的单一加速比，因为实测增益同时取决于模型架构、模拟晶胞和分子动力学集成。对于 NequIP-OAM，与 OpenEquivariance 在同一

CUDA 图包装器内的比较直接体现了这种对集成方式的依赖（第 3.3 节及补充信息 S4.5）。

4 结论

通过将融合卷积核与已部署径向多层感知机的三次 Hermite 制表相结合，在不存储或重新计算力的反向传播中的逐边 MLP 中间值的情况下，JET-MLIPs 加速了分子动力学，并降低了来自五个等变 MLIP 系列的 15 个已发布检查点的 GPU 内存占用。在每个模型现有单精度浮点数后端所设定的精度预算内，JET-MLIPs 内核 + 表在所有相对于 fp64 参考值进行测试的模型中均保持了单点力精度，同时将独立的 fp64 插值误差控制在远低于 fp32 舍入误差和多次运行间变化的水平。在分子动力学中，它保持了微正则系综能量守恒，并使统计观测量处于重复运行的波动范围内（第 3.1节）。唯一的例外是 NequIP-OAM L 中约为 6×10^{-7} eV 原子$^{-1}$的可忽略能量偏移；该偏移在 fp64 下消失，且不改变力。在一块 H200 上，不使用 D3 的 5632 原子电解质的 LAMMPS 分子动力学运行速度比现有的 TP 加速后端快 1.5–5。9× 倍，同时将全程峰值 GPU 内存降低了 2.4–24×，其中 SevenNet 的内存比代表下界（第 3.2 节和第 3.5节）。结合 JET-D3 后，这些提升使得在单个图形处理器上对约 10^4原子的液态和固态电解质进行常规的多纳秒输运计算成为可能（第 3.6节）。

在第 3.7 节所述的范围内，这些结果表明，消除径向多层感知机及其逐边中间值，而非仅加速张量积，可同时缓解采用 NequIP 卷积的模型在运行时间和内存方面的瓶颈。仅内核对照直接分离出这一贡献，适用于 SevenNet；而其他四个系列的端到端加速则结合了内核融合、径向制表和分子动力学集成。DPA4C 通过协同设计的架构解决了这一径向权重瓶颈，[31] JET-MLIPs 则无需重新训练，即可为现有预训练检查点解决这一问题。通过降低轨迹计算成本和内存需求，JET-MLIPs 大幅缓解了等变机器学习原子间势模拟的

GPU 瓶颈，并让大规模纳秒级分子动力学模拟能够在广泛可用的硬件上运行，包括消费级 GPU。

数据和软件可用性

JET-MLIPs（包含内核库） `jet_mlip`、SevenNet、MACE、NequIP、Nequix 和 EquFlash 的适配器、JET-D3 以及 LAMMPS 构建脚本，可通过以下链接获取 https://github.com/yanselmo-jk/jet-mlips ，并采用知识共享署名-非商业性使用-相同方式共享 4.0 国际许可协议（CC BY-NC-SA 4.0）授权。源自 LAMMPS 的文件（JET-D3 pair style、对 LAMMPS ML-IAP 的补丁，以及经 SevenNet 补丁修改的 LAMMPS 对样式）仍受 LAMMPS 的 GNU 通用公共许可证第 2 版约束。使用 JET-MLIPs 获得研究结果并发表的用户，请引用本文。发布的软件包未经修改使用或直接打补丁；GGNN（EquFlash）以加载而非复制的方式使用，且不重新分发任何模型权重。JET-MLIPs 的开发独立于 SevenNet、MACE、NequIP、Nequix 和 EquFlash 的开发者。基准测试输入、测量脚本和原始数据可向作者索取。

补充信息

测试体系和基准测试协议、内核实现和 LAMMPS 集成、精度测试和分子动力学验证、性能、内存和分子动力学集成的详细信息、应用示例、支持的布局和限制、表 S1–S7 和图 S1–S9（PDF）。

作者贡献

J.Y. 构思了这项研究，并设计和实现了融合内核、径向表、JET-D3 和 LAMMPS 集成。J.Y. 进行了基准测试、精度

和分子动力学验证及输运模拟，分析了数据并制作了图表。J.Y. 和 S.J. 撰写了稿件。两位作者均审阅并批准了最终版本。

备注

作者声明不存在相互竞争的经济利益。

致谢

本研究未获得外部资助。代码开发过程中使用了 Claude Fable 5.1 (Anthropic) 的协助，本文写作过程中使用了 Claude Fable 5.2 (Anthropic) 和 Gemini 4 Neon (Google) 的协助。作者审查并核实了所有代码、数据和文本，并对本文内容承担全部责任。

参考文献

(1) Behler, J.; Parrinello, M. 高维势能面的广义神经网络表示。*Phys. Rev. Lett.* **2007**, *98*, 146401.

(2) Bartók, A. P.; Payne, M. C.; Kondor, R.; Csányi, G. 高斯近似势：无需电子即可实现量子力学精度。*Phys. Rev. Lett.***2010**，*104*，136403。

(3) Behler, J. 高维神经网络势的四代发展。*Chem.Rev.***2021**，*121*，10037–10072。

(4) Unke, O. T.；Chmiela, S.；Sauceda, H. E.；Gastegger, M.；Poltavsky, I.；Schütt, K. T.；Tkatchenko, A.；Müller, K.-R. 机器学习力场。*Chem. Rev.* **2021年**, *121*，10142–10186。

(5) Schütt, K. T.；Kindermans, P.-J.；Sauceda, H. E.；Chmiela, S.；Tkatchenko, A.；Müller, K.-R. SchNet：用于建模量子相互作用的连续滤波卷积神经网络。Adv. Neural Inf. Process. Syst. 2017年。

(6) Xie, T.; Grossman,J. C. 用于准确且可解释地预测材料性质的晶体图卷积神经网络。*Phys. Rev. Lett.***2018**, *120*, 145301.

(7) Deng, B.; Zhong, P.; Jun, K.; Riebesell, J.; Han, K.; Bartel, C. J.; Ceder, G. CHGNet：一种用于电荷信息驱动原子建模的预训练通用神经网络势。***Nature Machine Intelligence*** **2023**, *5*, 1031–1041.

(8) Chen, C.; Ong, S. P. 一种适用于元素周期表的通用图深度学习原子间势。*Nat. Comput. Sci.* **2022**, *2*,718–728.

(9) Yang, H.; Hu, C.; Zhou, Y.; Liu, X.; Shi, Y.; Li, J., 等。MatterSim：一种适用于不同元素、温度和压力的深度学习原子模型。2024; arXiv:2405.04967.

(10) Neumann, M.；Gin, J.；Rhodes, B.；Bennett, S.；Li, Z.；Choubisa, H.；Hussey, A.；Godwin, J. Orb：一种快速、可扩展的神经网络势。2024；arXiv：2410.22570。

(11) Batzner, S.；Musaelian, A.；Sun, L.；Geiger, M.；Mailoa, J. P.；Kornbluth, M.；Moli- nari, N.；Smidt, T. E.；Kozinsky, B. 面向高效数据利用和高精度原子间势的 E(3) 等变图神经网络。《自然 · 通讯》 **2022**，*13*，2453。

(12) Musaelian, A.；Batzner, S.；Johansson, A.；Sun, L.；Owen, C. J.；Kornbluth, M.；Kozin- sky, B. 面向大规模原子动力学学习局部等变表示。《自然·通讯》 **2023**，*14*，579。

(13) Batatia, I.; Kovács, D. P.; Simm, G. N. C.; Ortner, C.; Csányi, G. MACE：用于快速、准确力场的高阶等变消息传递神经网络。神经信息处理系统进展 35。2022；第 11423–11436 页。

(14) Ju, S.; You, J.; Kim, G.; Park, Y.; An, H.; Han, S. 预训练通用机器学习原子间势在锂离子电池液态电解质物理化学模拟中的应用。*Digital Discovery* **2025**，*4*，1544–1559。

(15) Yoo, J. 使用机器学习分析和阐明先进二次电池体系中材料性质的研究。博士学位论文，首尔国立大学材料科学与工程系，2025。

(16) Ding, J.; Zichi, L.; Carli, M.; Wang, M.; Musaelian, A.; Xie, Y.; Kozinsky, B. 控制固态电池界面演化的耦合反应与扩散。2025；arXiv:2506.10944。

(17) Jia, W.; Wang, H.; Chen, M.; Lu, D.; Lin, L.; Car, R.; E, W.; Zhang, L. 借助机器学习，将具有从头算精度的分子动力学扩展至 1 亿个原子。SC20：国际高性能计算、网络、存储与分析会议。2020年；第 1–14 页。

(18) Kozinsky, B.; Musaelian, A.; Johansson, A.; Batzner, S. 将深度等变模型的领先精度扩展至真实规模的生物分子模拟。国际高性能计算、网络、存储与分析会议（SC23）论文集。2023年；第 1–12 页。

(19) Lu, D.; Wang, H.; Chen, M.; Lin, L.; Car, R.; E, W.; Jia, W.; Zhang, L. 采用从头算精度对 1 亿个原子进行 86 PFLOPS 深度势分子动力学模拟。计算物理通讯 **2021**，*259*，107624。

(20) Fan, Z.; Wang, Y.; Ying, P.; Song, K.; Wang, J.; Wang, Y. 等。GPUMD：用于构建精确机器学习势并执行高效原子模拟的软件包。《化学物理学杂志》 **2022**，*157*，114801。

(21) Riebesell, J.；Goodall, R. E. A.；Benner, P.；Chiang, Y.；Deng, B.；Ceder, G.；Asta, M.；

Lee, A. A.; Jain, A.; Persson, K. A. 评估机器学习晶体稳定性预测的框架。*Nature Machine Intelligence* **2025**, *7*, 836–847。

(22) Jacobs,R.; Morgan,D.; Attarian, S.; Meng, J.; Shen, C.; Wu, Z.; Xie, C. Y.; Yang, J. H.; Artrith, N.; Blaiszik, B.; Ceder, G.; Choudhary, K.; Csányi, G.; Cubuk, E. D.; Deng, B.; Drautz, R.; Fu, X.; Godwin, J.; Honavar, V.; Isayev, O.; Johansson, A.; Kozinsky, B.; Martiniani, S.; Ong, S. P.; Poltavsky, I.; Schmidt, K. J.; Takamoto, S.; Thompson, A.; Westermayr, J.; Wood, B. M. 机器学习原子间势实用指南——现状与未来。*Current Opinion in Solid State and Materials Science* **2025**, *35*, 101214。

(23) Park, Y.; Kim, J.; Hwang, S.; Han, S. 分子动力学模拟中图神经网络原子间势的可扩展并行算法。*Journal of Chemical Theory and Computation* **2024**, *20*, 4857–4868。

(24) Tan, C. W.; Descoteaux, M. L.; Kotak, M.; de Miranda Nascimento, G.; Kavanagh, S. R.; Zichi, L.; Wang, M.; Saluja, A.; Hu, Y. R.; Smidt, T.; Johansson, A.; Witt, W. C.; Kozinsky, B.; Musaelian, A. 深度等变原子间势的高性能训练与推理。*Digital Discovery* **2026**, *5*, 1558–1567。

(25) Geiger, M.; Smidt, T. e3nn：欧几里得神经网络。2022；arXiv:2207.09453。

(26) Lee, S. Y.; Kim, H.; Park, Y.; Jeong, D.; Han, S.; Park, Y.; Lee, J. W. FlashTP：用于机器学习原子间势的融合式稀疏性感知张量积。第42届国际机器学习会议论文集。2025；第33143–33156页。

(27) Oh, S.; You, J.; Kim, J.; Lee, J.; An, H.; Han, S.; Kang, Y. 通过知识蒸馏实现可扩展原子模拟的轻量级通用机器学习原子间势。*Journal of Chemical Information and Modeling* **2026**, *66*, 8875–8892。

(28) NVIDIA, NVIDIA cuEquivariance。 https://github.com/NVIDIA/cuEquivariance, 2024；访问日期：2026-10。

(29) Bharadwaj, V.; Glover,A.; Buluç, A.; Demmel, J. 面向 O(3)等变深度网络的高效稀疏内核生成器。2025 年应用与计算离散算法会议（ACDA）论文集。2025；第 32–46 页。

(30) Lu, D.; Jiang, W.; Chen, Y.; Zhang, L.; Jia, W.; Wang, H.; Chen, M. DP Compress：用于生成高效深度势模型的模型压缩方案。*Journal of Chemical Theory and Computation* **2022**，*18*，5559–5567。

(31) Li, T.; Xue, J.; Zhang, L.; Zhang, D.; Wang, H. 以经验势的速度实现通用机器学习分子动力学。2026；arXiv:2608.19041。

(32) Koker, T.; Kotak, M.; Smidt, T. E. 以有限预算训练材料基础模型。arXiv:2508.16067，2025。

(33) SamsungDS，GGNN：EquFlash 和 EquFlashV2 原子间势。 https://github.com/SamsungDS/GGNN，2026；访问日期：2026-11-01。

(34) Thompson, A. P.; Aktulga, H. M.; Berger, R.; Bolintineanu, D. S.; Brown, W. M.; Crozier, P. S.; in 't Veld, P. J.; Kohlmeyer, A.; Moore, S. G.; Nguyen, T. D.; Shan, R.; Stevens,M. J.; Tranchida,J.; Trott,C.; Plimpton, S. J. LAMMPS——一款用于原子尺度、介观尺度和连续尺度基于粒子的材料建模的灵活模拟工具。计算物理通讯 **2022**，*271*，108171。

(35) Larsen, A. H.; Mortensen, J. J.; Blomqvist, J.; Castelli, I. E.; Christensen, R.; Dułak, M.; Friis, J.; Groves, M. N.; Hammer, B.; Hargus, C.; Hermes, E. D.; Jennings, P. C.; Jensen, P. B.; Kermode, J.; Kitchin, J. R.; Kolsbjerg, E. L.; Kubal, J.; Kaasbjerg, K.; Lysgaard, S.; Maronsson, J. B.; Maxson, T.; Olsen, T.; Pastewka, L.; Peterson, A.; Rostgaard, C.; Schiøtz, J.; Schütt, O.; Strange, M.; Thygesen, K. S.; Vegge, T.; Vilhelmsen, L.; Walter, M.; Zeng, Z.; Jacobsen, K. W. 原子模拟环境——用于处理原子的 Python 库。《物理学：凝聚态物质》**2017年**，*29*，273002。

(36) Grimme, S.; Antony, J.; Ehrlich, S.; Krieg, H. 针对 94 种元素 H–Pu 的密度泛函色散校正（DFT-D）的一致且精确的从头算参数化。《化学物理学杂志》**2010**，*132*，154104。

(37) Grimme, S.; Ehrlich, S.; Goerigk, L. 色散校正密度泛函理论中阻尼函数的影响。《计算化学杂志》2011，32，1456–1465。

(38) Thomas, N.; Smidt, T.; Kearnes, S.; Yang, L.; Li, L.; Kohlhoff, K.; Riley, P. 张量场网络：用于三维点云的旋转与平移等变神经网络。2018；arXiv:1802.08219。

(39) Fey, M.; Lenssen, J. E. 使用 PyTorch Geometric 进行快速图表示学习。2019；arXiv:1903.02428。

(40) Lysogorskiy, Y.; van der Oord, C.; Bochkarev, A.; Menon, S.; Rinaldi, M.; Hammerschmidt, T.; Mrovec, M.; Thompson, A.; Csányi, G.; Ortner, C.; Drautz, R. 原子簇展开（PACE）的高性能实现及其在铜和硅中的应用。《npj计算材料》2021，7，97。

(41) Birkhoff, G.; Priver, A. 导数的Hermite插值误差。《数学物理学杂志》**1967**，*46*，440–447。

(42) Goldberg, D. 每位计算机科学家都应了解的浮点运算知识。*ACM Comput. Surv.* **1991**，*23*，5–48。

(43) Demmel, J.; Nguyen, H. D. 并行可复现求和。*IEEE Trans. Comput.***2015**，*64*，2060–2070。

(44) Shanmugavelu, S.; Taillefumier, M.; Culver, C.; Hernandez, O.; Coletti, M.; Sedova, A. 浮点非结合性对高性能计算和深度学习应用中可复现性的影响。SC24-W：国际高性能计算、网络、存储与分析会议研讨会。2024；第170–179页。

(45) Allen, M. P.; Tildesley, D. J. 《液体的计算机模拟》，第2版；牛津大学出版社，2017年。

(46) Stoddard, S. D.; Ford, J. Lennard-Jones气体系统的随机行为数值实验。《物理评论A辑》1973年，8，1504–1512。

(47) Posch, H. A.; Hoover, W. G. 致密Lennard-Jones流体的Lyapunov不稳定性。《物理评论A辑》1988年，38，473–482。

(48) Fu, X.; Wood, B. M.; Barroso-Luque, L.; Levine, D. S.; Gao, M.; Dzamba, M.; Zit-nick, C. L. 学习用于物理性质预测的平滑且富有表现力的原子间势。第42届国际机器学习会议论文集。2025；第17875–17893页。

(49) Johansson, A.; Weinberg, E.; Trott, C.; McCarthy, M.; Moore, S. LAMMPS-KOKKOS：跨百亿亿次级架构的可移植性能分子动力学。SC '25研讨会论文集。2025；第1217–1232页。

(50) Kraft, M. A.; Culver, S. P.; Calderon, M.; Böcher, F.; Krauskopf, T.; Senyshyn, A.; Dietrich, C.; Zevalkink, A.; Janek, J.; Zeier, W. G. 晶格极化率对锂超离子硫银锗矿 Li_6PS_5X（X ＝ Cl, Br, I）中离子电导率的影响。《美国化学会志》2017年，139，10909–10918。

(51) He, X.; Zhu, Y.; Epstein, A.; Mo, Y. 从头算分子动力学模拟所得扩散性质的统计方差。《npj计算材料》2018，4，18。

(52) Kim, J.; You, J.; Park, Y.; Lim, Y.; Kang, Y.; Kim, J.; Jeon, H.; Ju, S.; Hong, D.; Lee, S. Y.; Choi, S.; Kim, Y.; Lee, J. W.; Han, S. 优化通用机器学习原子间势的跨领域迁移。《自然·通讯》2026，17，3432。

(53) Moganty, S.; Abbate, L.; Raghunathan, A.; Sinicropi, J.; Torres,G.; Wu,Y. 含磷电解质。美国专利10,665,899 B2，NOHMs Technologies Inc.，2020年。

(54) Lee, J.; Ju, S.; Hwang, S.; You, J.; Jung, J.; Kang, Y.; Han, S. 采用机器学习势研究 Li_6PS_5Cl 中无序依赖的锂扩散。*ACS* 应用材料与界面 **2024**，*16*，46442–46453。

(55) Jeon, T.; Cha, G. H.; Jung, S. C. 理解控制硫银锗矿型 Li_6PS_5Cl 固态电解质中锂分布和扩散的阴离子无序。*J. Mater. Chem.A* **2024**，*12*，993–1002。

(56) Chen, J.; Fang, M.; Wu, Q.; Tang, S.; Zheng, J.; Wei, C.; Cao, X.; Shi, Y.; Xu, N.; He, Y. 通过机器学习势深入理解 Li_6PS_5Cl 中锂离子扩散的原子机制。*Chem. Mater.* **2025**，*37*，591–599。

(57) Yu, C.; Ganapathy, S.; Hageman, J.; van Eijck, L.; van Eck, E. R. H.; Zhang, L.; Schwietert, T.; Basak, S.; Kelder, E. M.; Wagemaker, M. 面向硫银锗矿型 Li_6PS_5Cl 固态电解质最佳结构–电导特性的简易合成。*ACS Appl. Mater. Interfaces* **2018**，*10*，33296–33306。

(58) Schlenker, R.; Hansen, A.-L.; Senyshyn, A.; Zinkevich, T.; Knapp, M.; Hupfer, T.; Ehrenberg, H.; Indris, S. 通过中子衍射、对分布函数分析和核磁共振研究 Li_6PS_5Cl 硫银锗矿型中的结构与扩散路径。《化学材料》**2020年**，*32*，8420–8430。

(59) Lysogorskiy, Y.; Bochkarev, A.; Drautz, R. 用于基础机器学习原子间势的图原子簇展开。《*npj*计算材料》**2026**，*12*，114。

(60) Xu, Z.; Xie, W.; Hu, P. 笛卡尔空间中具有通用嵌入的谱/空间张量原子簇展开。2025；arXiv:2509.14961。

(61) Liao, Y.-L.; Hoffman, A. J.; Shen, S. C.; Duval, A.; Norwood, S. W.; Smidt, T. EquiformerV3：用于高效、富有表现力且通用的 SE(3) 等变图注意力变换器的尺度扩展。2026; arXiv:2604.09130.

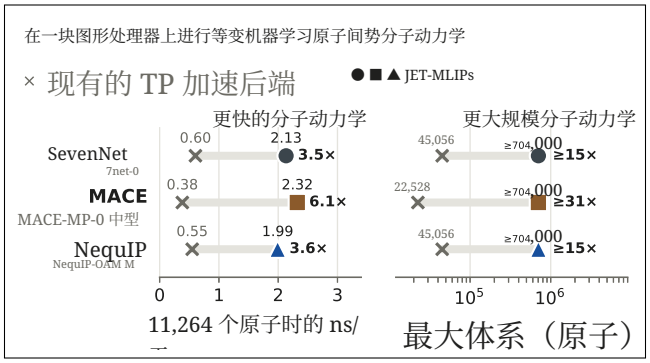
(62) Xu, Z.; Xie, W.; Hu, P. 用于原子间势的径向旋转注意力边缘簇展开。2026; arXiv:2607.10664.

(63) Kairos Materials, Prophet：跨组成、构型和自旋扩展原子级基础模型。 https://www.kairosmaterials.com/papers/Prophet.pdf，2026；技术报告；访问于 2026-11-01。

(64) Thiemann, F. L.; Reschützegger, T.; Esposito, M.; Taddese, T.; Olarte-Plata, J. D.; Martelli, F. 通过自回归等变网络实现无力分子动力学。*Nature Machine Intelligence* **2026**，*8*，764–776。

(65) Liao, Y.-L.; Wood, B.; Das, A.; Smidt, T. EquiformerV2：改进的等变 Transformer，实现向高阶表示扩展。国际学习表征会议（ICLR）。2024。

目录图


在一块图形处理器上进行等变机器学习原子间势分子动力学
× 现有的 TP 加速后端
● ■ ▲ JET-MLIPs
更快的分子动力学
更大规模分子动力学
SevenNet
7net-0
MACE
MACE-MP-0 中型
NequIP
0.60
2.13
3.5×
0.38
2.32
6.1×
0.55
1.99
3.6×
45,056
≥704,000
≥15×
22,528
≥704,000
≥31×
45,056
≥704,000
≥15×
0
1
2
3
10^5
10^6
11,264 个原子时的 ns/
最大体系（原子）

支持信息
通过径向制表和内核融合加速分子动力学的等变原子间势

Jaekyun Yoo (worbs3237@gmail.com) 和 Suyeon Ju (suyeonju224@gmail.com) 独立研究者

S1 测试体系与基准测试协议

本节介绍第 2.6 节的测试体系和基准测试条件，其中一个化学结构完整的 5632 原子电解质晶胞用于计时基准测试和分子动力学验证。

S1.1 测试体系

MD 基准测试的常见工作负载是$_6$ 碳酸乙烯酯（EC）和碳酸甲乙酯（EMC）中的 LiPF[1] 溶液，该体系此前已使用机器学习力场进行建模[1] （正文第 2.6 节和表 2）。每个后端接收相同的原子和边数。主晶胞包含 5632 个原子（64 个 $LiPF_6$, 128 EC 和 256 个 EMC，即 Li 64、P 64、F 384、C 1408、O 1152 和 H 2560）。具有相同组成的 352 原子晶胞（4 个 $LiPF_6$, 8 EC 和 16 个 EMC，立方晶胞，边长为 $L = 15.37$Å）用于精度测试，因为在 5632 个原子时，其 fp64 参考值超出图形处理器内存。该晶胞还用于 EquFlash V1 的小晶胞计时测试（第 S4.5节）。

主晶胞由完整分子堆积而成，并使用带有 Perdew–Burke–Ernzerhof（PBE）参数的 7net-0 加 D3(BJ) 进行 200 ps 的 NPT分子动力学平衡，能量的截断值为 9000 $bohr^2$，配位数的截断值为 1600$bohr^2$。平衡后密度为 1.361(7) g cm^{-3} ($L = 38.74$ Å)，所有 384 个溶剂分子和 64 个 PF_6^-阴离子均保持完整。计时运行从在 300开尔文 下使用 7net-0（仅 JET-MLIPs 内核）加 D3 进行额外 10 ps 的 NVT系综后达到的状态开始，步长为 0.5 fs。

为测试不同系统尺寸和材料类别下的尺度扩展，规模扫描采用固定晶胞的周期性复制体。电解质扫描将 5632 原子晶胞复制 2 至 512 次，原子数从 11,264 增至 2,883,584（例如，45,056 个原子对应 $2 \times 2 \times 2$ 份，152,064 个原子对应 $3 \times 3 \times 3$ 份）。$LiCoO_2$晶胞沿层状结构的正交 24 原子晶胞重复 $5\times3\times1$、 $6\times4\times2$、 $10\times6\times2$和 $13\times8\times2$次（分别为 360、1152、2880 和 4992 个原子），将 4992 原子晶胞复制 $2\times2\times2$次后得到 39,936 个原子。硅晶胞沿每个轴重复 8 原子立方金刚石晶胞 n 次，范围为 $n = 3$–32（216 至 262,144 个原子）。非晶硅氧化物晶胞沿每个轴重复 70 原子晶胞（1 nm 立方体中含 23 个硅原子和 47 个氧原子，使用 7net-0 熔融淬火） $n = 1$–10 次，原子数从 70 增至 70,000。

S1.2 硬件和软件

所有主要计时数据均在配备三块 NVIDIA H200 NVL GPU 的单个节点上测得。每块 GPU 提供 141 GB 内存（nvidia-smi 报告为 143，771 MiB），具有 sm_90 计算能力，且

132 个流式多处理器（SM），运行驱动程序版本 595.99.02，且 功率上限为 600 W 。主机节点配备两颗 AMD EPYC 9654 处理器、251 吉比字节 内存和 Ubuntu 24.04。第 3.6 节主要部分中的输运计算是在 NVIDIA H200 或 RTX 2080 Ti GPU 上进行的。

SevenNet 的运行使用 Python 3.11、PyTorch 2.7。1+cu126 [4], CUDA 12.6、e3nn 0.6.0、SevenNet 0.13.0（JET-MLIPs 行由 JET-MLIPs 修补）、cuEquivariance 0.9。1 [5], OpenEquiv- ariance 0.7。0 [6] 以及 FlashTP 0.1。0 [7] 针对 sm 构建_90。其他系列使用 mace-torch 0.3.16、PyTorch 2.7.1 和 cuEquivariance 0.9.1（MACE）；NequIP 0.19.1、PyTorch 2.11.0 和 OpenEquivariance 0.7.0（NequIP-OAM）；nequix 0.4.5 及其 torch 后端和 OpenEquiv- ariance 内核（Nequix）；以及 GGNN（提交 `a25dfa4`），配合 PyTorch 2.9.1 和 cuEquivariance 0.6.0（EquFlash）。

这五个系列均使用 LAMMPS 机器学习原子间势接口（ML-IAP）[8]。通过 ML-IAP 进行的运行使用 LAMMPS 2025 年 7 月 22 日更新 6（SevenNet 文档注明为更新 4），并以 Kokkos 编译；Kokkos 为该接口提供可移植的并行执行和设备数据管理 [9]。对于 SevenNet 的 pair style `e3gnn`，LAMMPS stable 2023 年 8 月 2 日更新 3 使用相同源代码编译了两个版本，一个包含 JET-MLIPs 的 pair style，另一个包含上游 SevenNet 0.13.0 的 pair style。所有后端均禁用了 TF32（TensorFloat-32）矩阵运算，每次运行均使用一个 MPI 秩和一个 OpenMP 线程。

S1.3 运行协议与后端

本小节介绍正文第 2.6 节未涵盖的基准测试协议细节。各系列现有的 TP 加速后端定义于该节。对于 MACE，该后端是由 MACE 脚本导出的 cuEquivariance 0.9.1 模型 `mace_create_lammps_model`；对于 NequIP-OAM，则是通过 NequIP 的修饰器启用的 OpenEquivariance 0.7.0 `enable_OpenEquivariance`。其他 SevenNet 后端见 S4.1 节。在本补充信息（SI）中，“上游”始终指未经修改的 SevenNet 0.13.0 版本、其后端（e3nn、FlashTP、cuEquivariance 和 OpenEquivariance）及其 `pair_style d3`。相比之下，正文表2中的上游模型，是各系列未经加速的模型。

每次运行仅在没有 GPU计算进程处于活动状态，且 GPU利用率 已连续 3 秒保持在 3% 或以下时开始。在小晶胞、时间线、D3 和消融系列测试期间，节点上的另外两块 GPU 均处于空闲状态。在其他系列测试期间，它们运行了本研究的其他基准测试，但从未运行上游 D3，因为后者的主机负载会拖慢相邻运行。并发作业存在与否的重复运行结果相差不超过 1%。

LAMMPS 计时采用 500 步的微正则系综运行，时间步长为 0.5 飞秒，初始状态为 300开尔文，每 10 步输出一次热力学数据。在电解质中，步骤 50 至 500 之间的 45 个间隔中有 9–10 个包含邻居列表重建，而 SevenNet 硅和非晶硅 O_2 扫描采用了 460 步且不进行重建。H200 上的 LAMMPS 步耗时取三次运行的中位数。现有的 TP 加速和 JET-MLIPs 内核 + 表格列采用五次运行的数据，见主表 2、图 S9c 中的 cuEquivariance 柱状图，以及使用上游 D3 的运行结果。规模扫描报告每个构型的首次运行结果。因此，对于 5632 个原子的 7net-nano-5.5，扫描得出的相对于 cuEquivariance 的加速比为 5。8×，而五次运行中位数得出的结果为 5。7× 。对于 7net-nano-5.5 和 7net-0，在 5632 个原子以上规模的电解质扫描中进行了两次额外重复，结果证实无重建步耗时相对于中位数的变化幅度最多为 1.1%（JET-MLIPs）和 0.2%（cuEquivariance），使三次运行所得加速比的变化最多为 1.0%。

模型调用计时在每个后端和系统规模各自的单个进程中完成 10 次预热调用后，使用 CUDA事件进行测量。每次计时是该次运行中各调用的平均 GPU 时间，模型调用内存则是单次调用的峰值 PyTorch 分配。整块 GPU 峰值是最大内存

使用 nvidia-smi 报告的 GPU 占用内存，并在 500 步运行期间每 200 毫秒采样一次，再减去运行前的读数。它同时包括 CUDA 上下文和分配器预留内存。该指标用于规模扫描以及 `e3gnn` 运行，因为这些运行未记录 PyTorch 分配统计。对于现有的 TP 加速后端，由于保留的分配器缓存可以在不影响步时间的情况下释放，运行期间整块 GPU 读数会逐步增加。因此，跨系列的比较使用第 2.6节的全程峰值。

S2 内核实现与 LAMMPS 集成

基于第 2.1 节和第 2.2 节中的卷积方程与成本分析，本节详细介绍第 2.3 节的层形状、融合内核、第 2.3 节的径向表以及第 2.5 节的 LAMMPS 集成。

S2.1 张量积路径与层形状

本小节中的形状采用 SevenNet 的形状；第 2.2 节中的成本分析针对的正是这一系列。表S2 列出了其他系列的尺寸。

SevenNet 相互作用层。 在 SevenNet 相互作用层（主图 1a）中，主公式（1）中的卷积嵌入如下。SevenNet 模型嵌入每个原子的物种 Z_i，并应用 T 个相互作用层 $t = 1, \ldots\ldots, T$（$T = 3$ 或 5，表 S2）。记原子 i 经过第 t 层后的特征为 $x_i^{(t)}$，模型根据以下方式读出总能量：

$$x_i^{(0)} = \mathrm{Linear}_{\mathrm{embed}}\big(\mathrm{onehot}(Z_i)\big), \tag{1}$$

$$s_i = \mathrm{SC}_t\big(x_i^{(t-1)}, Z_i\big), \qquad y_i = \mathrm{Linear}_{\mathrm{pre},t}\big(x_i^{(t-1)}\big), \tag{2}$$

$$h_i = \text{卷积}_t\big(\{y_j, \mathbf{r}_{ij}\}_{j\in\mathcal{N}(i)}\big), \tag{3}$$

$$x_i^{(t)} = \mathrm{Gate}\big(z_i\big), \qquad z_i = \mathrm{Linear}_{\mathrm{post},t}(h_i) + s_i, \tag{4}$$

$$E = \sum_i E_i, \qquad E_i = \sigma_{Z_i}\,\mathrm{Linear}_E\big(\mathrm{Linear}_h(x_i^{(T)})\big) + \mu_{Z_i}. \tag{5}$$

每个 线性层 都是一种等变线性映射，分别在每个次数内混合通道；而 SC_t是依赖物种的线性自连接。在式（4）中，z_i 为每个 $l > 0$通道携带一个额外的门控标量。门 函数应用依赖宇称的激活函数（偶标量使用 SiLU，奇标量使用双曲正切），并用激活后的门控标量缩放每个 $l > 0$ 通道。由于最后一层仅输出标量，式（5）应用两个线性映射，随后进行依赖物种的缩放 σ_{Z_i} 和平移 μ_{Z_i}，以生成原子能量 E_i。NequIP-OAM 遵循相同的布局（主文图 1a），有三处不同。它的第一层省略了 SC_1, ，其读出层使用一个线性映射和一个 Ziegler–Biersack– Littmark（ZBL）成对项。此外，它在张量积之前应用邻居归一化 $1/\sqrt{N_{\mathrm{avg}}}$，这等价于主文公式（$\nu = \sqrt{N_{\mathrm{avg}}}$）中的 5）。

对于正文第2.2节分析的检查点，表S1 列出了每层的输入不可约表示、活动路径，N_W以及每条边的融合乘加次数。剪除未使用的输出后，7net-nano-5.5 的中间层 15 条路径中有 4 条被移除。单精度浮点数下，径向权重每条边占用 $4N_W$ 字节，因此 7net-nano-5.5（$N_W = 352$）、7net-0（$N_W = 960$）和 7net-omni（$N_W = 3008$）的中间层分别占用 1408、3840 和 12，032B 。在 5632 原子晶胞中，7net-nano-5.5 和 7net-0 的中间层权重总计为515MiB 和1036MiB (383,648 和282,982 条边，见表S2）。反向传播还会增加一个同等大小的 $\partial E/\partial \mathbf{W}$ 张量。

S2.2 融合前向与反向传播内核

算法 S1 和 S2 概述了默认的表格内核。表格模式和仅内核模式（仅 SevenNet 支持）都会为每个目标原子分配一个线程束，在按目标原子排序的压缩稀疏行（CSR）边索引上运行，并将线程束通道映射到 u 通道 u，以 32 个通道单元为一组。每个线程束以分块方式流式处理关联边，在共享内存中暂存 r 和 $Y(\hat{}\mathbf{r}ij)$，聚集 y_j 并应用非零的克莱布施–戈丹纤维（图 S1b、c）。它对每条路径和每个通道单元只乘一次径向权重，而不是对每个纤维都乘一次

算法 S1 单层的融合前向卷积。

输入： 按目标原子排序的边压缩稀疏行（CSR）索引 $j \to i$、边向量 $\mathbf{r}_{ij}$，特征 y_j 和径向表 $\mathcal{T}$（第 2.3节）

输出： 每个目标原子的卷积输出 h_i i

对于 每个路径组（补充信息 S2.3）和目标原子 i **执行**
  初始化 $\text{acc}_{p,\ u} \leftarrow 0$ ▷ 输出累加器、寄存器
  对于每条边 $j \to i$ 在 CSR 的第 i 行中**执行**
    加载 $\mathbf{r}_{ij}$ ▷ 全局内存到寄存器
    计算 $r \leftarrow |\mathbf{r}_{ij}|$ ▷ 共享内存
    计算 $Y(\hat{\mathbf{r}}_{ij})$ ▷ 共享内存
    加载 y_j ▷ 全局内存到寄存器
    通过 $\mathcal{T}$ 中的 Hermite 查找计算 $\mathbf{W}(r)$ ▷ 全局表到寄存器
    将 $W_{p,u}[y_j \otimes Y]_p$ 累加到 $\text{acc}_{p,u}$ ▷ 式（4），寄存器
  结束循环
  为该组的列写入 $h_i \leftarrow \text{acc}/\nu$ ▷ 全局内存，单个写入者
结束循环

。每通道累加器在遍历邻居列表期间始终保存在寄存器中，最后一次性写入且不使用原子操作，因此前向传播具有确定性。

在仅内核模式下，前向内核会以寄存器内计算公式 (2) 和 (3a,b) 替代表查找。此计算使用精确的超越函数内建指令（不采用快速数学近似），并针对存放在共享内存中的多层感知机权重执行通道并行的矩阵–向量乘积。反向传播内核会重新计算边嵌入，并在寄存器中形成 $\partial E/\partial W$ 。仅内核模式会将此梯度直接经由多层感知机传播至 $\partial E/\partial \mathbf{r}_{ij}$ ，并通过逐边原子操作完成。相比之下，表格模式会从插值函数中读取 $\mathbf{W}'(r)$ ，并在内核 2 中对每条边应用一次调和函数雅可比矩阵（算法 S2，第 S2.3 节）。在两种模式下， $\partial E/\partial y_j$ 都会按线程束暂存于共享内存中，再通过合并原子操作累加到全局内存；这会引入主文第 3.1 节测得的不同运行间力的变化。

在仅内核模式下，对于多组层，部署标志 `--exact_backward` 用于选择两种变体之一。`recompute` 变体会在每个组中重复执行多层感知机反向传播，无需逐边缓冲区。默认值 `split` 变体则通过原子操作缓冲 $\partial E/\partial \mathbf{a}_2 \in \mathbb{R}^{64}$ ，每条边占用 256B ；对于包含 5632 个原子的 7net-nano-5.5 中间层，这相当于 93.7MiB 。随后，第二个内核会在释放缓冲区前，对每条边完成一次多层感知机反向传播（第 S4.7 节讨论内存权衡）。

S2.3 路径组与构建时特化

路径组是指一个线程束在遍历其原子边的过程中，一次处理的通道单元集合。每个通道单元都是一条 TP路径，限定为连续的 32 个通道，每个线程束通道对应一个通道。少于 32 个通道的路径占用一个通道单元，其余通道将处于空闲状态。每次构建内核时，都会将层形状划分为若干组，使其满足目标图形处理器的共享内存和寄存器限制。特化在构建时完成：此时，每个层形状都会转换为编译时的路径表和纤维表，其中仅保留每条路径的非零 Clebsch–Gordan项。随后，纤维循环会被完全展开，累加器则保留在寄存器中。

表格内核不包含 MLP 状态，因此其表组大小由插值、谐波和累加器所需的寄存器数量决定。每增加一个表组，就会重新流式处理一个原子的关联边，并重复进行几何计算和插值计算。表格

算法 S2 单层力的反向传播融合。

输入： 算法 S1 的输入 $\partial E/\partial h_i$
输出： $\partial E/\partial y_j$ 和 $\partial E/\partial \mathbf{r}^{ij}$

初始化 $\partial E/\partial y_j \leftarrow 0$ ▷ 全局内存
初始化 $\mathcal{G}_{ij} \leftarrow 0$ ▷ 逐边缓冲区，用于（g_Y， g_r），全局内存
内核 1
对于每个路径组和目标原子 i **执行**
  加载 $\partial E/\partial h_i$ ▷ 全局内存到寄存器
  将上游梯度除以 ν ▷ 寄存器
  遍历每条边 $j \to i$，在压缩稀疏行的第 i 行中**执行**
    加载 $\mathbf{r}^{ij}$ ▷ 从全局内存加载到寄存器
    计算 $r \leftarrow |\mathbf{r}_{ij}|$ ▷ 共享内存
    计算 $Y(\hat{\mathbf{r}}^{ij})$ ▷ 共享内存
    加载 y_j ▷ 将全局内存加载到寄存器
    通过在 $\mathcal{T}$ 中进行 Hermite 查找来计算 $\mathbf{W}(r)$ ▷ 将全局表加载到寄存器
    从同一个插值函数计算$\mathbf{W}'(r)$ ▷ 径向导数、寄存器
    通过张量并行反向传播计算 g_y ▷ 组的特征梯度、寄存器
    通过张量并行反向传播计算 g_Y ▷ 组的调和梯度、寄存器
    计算 g_W $_{p,u}$ 通过张量并行反向传播 ▷ 组的权重梯度，寄存器
    将 g_y 累积到 $\partial E/\partial y_j$ ▷共享暂存，全局原子操作
    计算 $g_r \leftarrow \sum_{p,u} g_{W_{p,u}} W'_{p,u}$ ▷ 组的径向梯度、寄存器
    将 (g_Y, g_r) 累加到 $\mathcal{G}_{ij}$ ▷全局原子操作，组累加
  结束循环
结束循环
内核 2
对 压缩稀疏行中的每条边 $j \to i$ 执行**do**
  加载 $\mathbf{r}^{ij}$ ▷ 从全局内存加载到寄存器
  计算 $r \leftarrow |\mathbf{r}_{ij}|$ ▷ 寄存器
  加载（g_Y， g_r）$\leftarrow \mathcal{G}_{ij}$ ▷ 全局内存到寄存器
  写入 $\partial E/\partial \mathbf{r}_{ij} \leftarrow (\partial Y/\partial \mathbf{r}^{ij})^\top g_Y + g_r \hat{\mathbf{r}}^{ij}$ ▷ 全局内存，单个写入者
结束循环

反向传播因此只计算每个表组中处于活动状态的寄存器，并且当减少表组数量所带来的收益超过寄存器溢出的代价时，允许略微超过 255 个寄存器上限。因此，7net-nano-5.5 的每一层在两个方向上都能装入一个表组。在 H200 上，7net-omni 最宽的层（64 条路径，$N_W = 3008$）在表格模式下的反向传播使用 8 个表格内核组，而仅内核模式下使用 32 个（或按仅内核寄存器计数时使用 94 个）。表格反向传播通过大小为 $[(l_{\max}+1)^2+1] \times 4\,\mathrm{B}$ 的临时逐边缓冲区，将调和函数雅可比矩阵卸载到内核 2；其中，$l_{\max} = 3$ 对应 68 B，$l_{\max} = 2$对应 40 B。反向传播完成后会释放该缓冲区，因此它只会在 7net-omni 的表调用中增加峰值内存。

S2.4 表格化径向权重

该表使用均匀节点 $r_k = kh$，其间距为 $h = r_c/(N'-1)$，其中 N' 是下文所述对齐后的节点数。对于每个节点和每一列 c，每个条目存储四个单精度浮点值，即 $W_c(r_k)$、$W'_c(r_k)$、差商 $(W_c(r_{k+1}) - W_c(r_k))/h$ 以及 $W'_c(r_{k+1})$。内核通过一次 16 字节加载读取一个条目，每层占用 $N' \times 4N_W \times 4\,\mathrm{B}$。

正文第 2.3 节介绍了三项数值保护措施。首先，差商（W（r_{k+1}） –W（r_k））$/h$ 预先以双精度浮点数计算，从而避免消去误差；否则，该误差会使 $N = 8192$ 处的最大力误差升高至 1.4×10^{-3} eV Å^{-1}。其次，将一个节点与截断起始点 r_{on} 对齐；对于 7net-nano-5.5、7net-0 和 7net-omni，该起始点分别为 5.0、4.5 和 5.5 埃。为此，N' 被设置为（$N/2$、N]）中的最大整数，使某个节点位于 r_{on} 上，从而将所需的 $N = 2048$ 降至 2047、2041 和 2041。第三，$(N' - 1)/r_c$ 被存储为两个单精度浮点数之和。这消除了边长最高达 6×10^{-8} 的相对舍入误差；该误差会引入系统性能量偏差，但不会改变力。

S2.5 与 LAMMPS 集成

JET-MLIPs 通过一个由五个系列共用的 ML-IAP 封装器连接到 LAMMPS [10]。对于 SevenNet，它还提供 pair style `e3gnn`，该样式仍是 SevenNet 的主要后端（主要内容见第 2.5 节，两个接口的比较见第 S4.5 节）。下文介绍图重放、GPU 边构建和系列适配器，第 S4.2 节介绍 JET-D3。

CUDA 图重放。 CUDA 图通过一次启动重放已记录的 GPU 操作序列，从而降低 CPU 启动开销 [11]。`pair_style e3gnn` 步包括边构建、模型前向计算、$\partial E/\partial \mathbf{r}$ 以及力和维里归约。它针对局部原子数与边分桶的每种组合捕获一次，之后进行重放。尽管边数会向上取整到某个分桶大小，目标 CSR 会在实际边数处终止，从而避免内核处理填充条目。下文介绍的 ML-IAP 封装器也会以相同方式重放其模型调用。

GPU 边构建。 在 `e3gnn`中，LAMMPS 的完整邻居列表仅在重新构建时上传。在每一步中，图内的三个固定形状内核会根据当前位置选出截断半径内的配对，并构建目标 CSR。它们使用显式舍入的双精度浮点数距离平方，且不进行 FMA 收缩，从而生成与 LAMMPS 主机循环逐位相同的边集合。如果边数超过捕获容量，则会调整缓冲区大小、重新捕获图，并重新计算该步。

ML-IAP 封装器和系列适配器。 该表存储完整的 $\mathbf{W}(r)$，包括截断半径和基函数归一化，因此内核无需依赖各代码构建径向基的方式。第 2.5 节的系列适配器会原地修补已安装的软件包（SevenNet、MACE），或重写已加载的模型（NequIP、Nequix、EquFlash）。MACE 内核之外的模块使用 PyTorch 编译器 `torch.compile`进行编译。相比之下，MACE 发布的 ML-IAP 模型（该模型 `mace_create_lammps_model` 使用 cuEquivariance 创建）以急切模式运行，不进行编译或 CUDA 图捕获。NequIP 适配器会直接向内核提供原始边向量，而不是预先计算球谐函数。

单个 ML-IAP 封装器即可部署包括 SevenNet 在内的所有系列。它是 LAMMPS 端独立于适配器的模块，要求每个系列仅提供其模型前向计算、截断半径以及索引和向量类型。在单个进程上，该封装器会将每个幽灵原子折叠到其所属原子中，从而消除层间幽灵特征的交换。它会从卷积 CSR 中屏蔽容量填充，并将包括输入准备在内的整个模型调用作为单个 CUDA 图进行重放。EquFlash V2 不执行图重放，因为在 cuEquivariance 0.6.0 和 0.9.1 下，cuEquivariance 启动都会使流捕获失效。在多个 MPI 进程上，所有系列都会退回到逐层幽灵交换，不进行幽灵原子折叠或图重放。这种

交换使用已发布的 SevenNet、MACE、NequIP-OAM 和 EquFlash 封装器；对于 Nequix，则使用 JET-MLIPs 的逐层交换钩子。

S3 精度与分子动力学验证

S3.1 参考测试与完整精度表

本小节详细介绍了支持正文第 3.1 节的参考测试、fp64 比较和重复调用检查。除非另有说明，精度测试均采用第 S1.2 节所述的硬件和软件，并禁用 TF32。测试使用了 7net-nano-5.5 在 300 开尔文 下进行的 NVE 轨迹中相隔 0。5 皮秒 的十帧 5632 原子电解质构型。对于 7net-omni [12], NequIP-OAM L 和 EquFlash，由于它们在 5632 个原子下进行 fp64 评估所需的内存超过图形处理器内存，因此改用 352 原子晶胞的十帧（第 S1.1 节）。SevenNet 接受了最全面的验证，因为它除仅内核模式外，还提供多个 fp32 后端（e3nn、FlashTP、cuEquivariance 和 OpenEquivariance）；其 ASE 接口以 fp64 构造边向量，因此单点 fp32 舍入下限最低。对于其余系列，将 JET-MLIPs 内核 + 表与已发布的 fp32 TP 加速模型和 fp64 模型进行比较（表S3）。

参考测试。对公式（2）–（5）的纯 PyTorch 实现，在 7net-nano-5.5 和 7net-0 的每一层上，均与 SevenNet 的 e3nn 模块匹配，误差为 10^{-10} （双精度浮点数）。以双精度浮点数实例化的仅内核 CUDA 内核，在相同容差下通过了相对于该参考实现的前向测试。在双精度浮点数下，对于 7net-nano-5.5 和 7net-omni，内核复现 e3nn 力的误差为 $1.7 \times 10^{-14}\,\mathrm{eV}$埃$^{-1}$；对于 7net-0 及其多保真对应模型 7net-mf-0 [13]（在其 PBE 模态下评估），误差为 $(1.3\text{–}2.8) \times 10^{-8}\,\mathrm{eV}$ 埃$^{-1}$。尚未确定后两种模型差异较大的原因。

fp32 实现之间的差异。与 e3nn 直接进行 fp32 比较时（5632 原子的电解质体系，7net-omni 为 352 个原子），JET-MLIPs 仅内核的最大力差为 (1.00–1.18) $\times 10^{-5}$ eV 埃$^{-1}$，是 FlashTP、cuEquivariance 和 OpenEquivariance 的 (2.3–6.4) $\times 10^{-6}$ eV 埃$^{-1}$的 1.7 至 4.3 倍。其能量与 e3nn 的差异在 1.7×10^{-8} eV/原子$^{-1}$以内，应力差异在 1.0×10^{-8} eV 埃$^{-3}$以内。然而，与双精度浮点数结果相比，JET-MLIPs 与 e3nn 的精度相当（表 S3）。e3nn 自身的 fp32 舍入误差最大约为 10^{-5} eV 埃$^{-1}$，与 JET-MLIPs 和 e3nn 之间的差异相当，因此直接比较 fp32 结果时，两种实现中的舍入误差都会产生影响，无法据此判断哪一种更准确。

相对于双精度浮点数的单点精度。表 S3 列出了主文第 3.1 节和主文表 1 所依据的均方根和最大力分量误差以及每原子能量误差；图 S2a、b 展示了 7net-nano-5.5 和 7net-omni 的力误差分布。对于 5632 原子晶胞的总能量（-3.2×10^4eV），一个单精度浮点数的最后一位单位（ulp） [14] 为 1.95×10^{-3} eV，或3.5×10^{-7} eV/原子$^{-1}$。在 SevenNet、NequIP-OAM 和 Nequix 中，均方根能量误差均接近这一分辨率极限。在 SevenNet 中，JET-MLIPs 内核+ 表的误差为 (1.0–3.4) $\times 10^{-7}$ eV/原子$^{-1}$，fp32 e3nn 的误差为 (0.93–3.3) $\times 10^{-7}$ eV/原子$^{-1}$。唯一的例外是 NequIP-OAM L 中 JET-MLIPs 内核 + 表存在近乎恒定的偏移，约为 -6×10^{-7} eV/原子$^{-1}$（1.7 最后一位单位）。使用双精度浮点数计算该表时，这一偏移会消失，且力保持不变。在 MACE 和 EquFlash 中，对原子能量进行单精度求和会使基线能量误差分别升至 80–800 最后一位单位和 5 最后一位单位，而 JET-MLIPs 的误差仍在现有后端误差的 1.01 倍以内。

对于 SevenNet 的力，FlashTP 的均方根误差与 fp32 e3nn 相当，最大误差相差不超过 8%。使用 D3 的 LAMMPS 运行以 fp64 e3nn 加 fp64 s-dftd3 [15] 作为参考（图 S2a、b）。与该参考相比，JET-MLIPs 内核 +表（搭配 JET-D3）和上游 e3nn（搭配上游 D3）的均方根误差均处于 fp32 e3nn 的 0.99–1.05 倍范围内，而这两种 D3

实现单独与 s-dftd3 相比的均方根差异为 4.4×10^{-8} eV Å^{-1}。即使在 352 个原子的情况下，EquFlash V2 也没有 fp64 参考，因此直接与其现有的 fp32 TP 加速后端进行比较。均方根力差和最大力差分别为 1.49×10^{-6} eV Å^{-1} 和 1.13×10^{-5} eV Å^{-1}，均方根能量差为 1.49×10^{-6} eV atom^{-1}，与 EquFlash V1 的 fp32 舍入误差相当。

在其余系列中，单点最大力误差更大（在 5632 原子晶胞中为 1.1×10^{-4} eV Å^{-1}，而 SevenNet 为 (1.2–1.5) $\times10^{-5}$ eV Å^{-1}）。这些误差源于各系列的 ASE 接口，该接口根据单精度浮点数位置构建边向量。跨越周期性边界时，减去大小为 L 的坐标会使边向量产生舍入误差，误差为 L 的半个最后一位单位（在 $L=38.7$ Å 处为 1.9×10^{-6} Å，在 $L=15.4$ Å 处为 4.8×10^{-7} Å）。在 MACE、NequIP-OAM 和 Nequix 中，这种舍入会使最大误差集中在跨越周期性边界且键长小于 1.75 Å 的原子上。在 5632 原子晶胞中，所有十帧的最大误差均出现在这类原子上，而其余原子中的最大误差小 2 至 4 倍。JET-MLIPs 接收相同的边向量，因此也出现相同的最大误差。相比之下，LAMMPS 接口根据双精度浮点数位置构建边向量，因此在分子动力学中不会出现这一伪差。

表误差与尺寸 N 的关系。对于三次 Hermite 表，插值误差的上界为 $h^4\max|W(4)_c|/384$ in $W_c(r)$，以及 $\sqrt{3}\,h^3\max|W(4)_c|/216$ in $W_c'(r)$。$\bar{N}$ 每翻倍一次，这些上界便分别缩小 16 倍和 8 倍；实测力误差也随之下降，直至达到 fp32 舍入下限（主文图 3a、b 和图 S3a、b）。SevenNet 和其他系列在 $N=1024$–2048 时均收敛至各自的 fp32 下限。在 SevenNet 中，$N=2048$–8192 范围内的误差分别保持在 fp32 e3nn 均方根误差的 0.94–1.06 倍，以及其最大误差的 0.85–1.22 倍。例外是 MACE-MP-0 medium：其均方根误差在 $N=2048$ 时为下限的 1.21 倍，并在 $N=4096$ 时收敛（表 S3）。表格化卷积时间从 $N=256$ 增至 8192 时增加了 6–15%。

由于编译表核仅支持 fp32，因此通过在 fp64 表上使用 PyTorch 应用相同的三次厄米公式来替代径向多层感知机，以 fp64 评估孤立表精度。在起始点对齐网格上，N每加倍，孤立表力误差降低八倍（图 S2c–e）。在 $N=2048$ 时，四个 SevenNet 模型的均方根误差达到 (0.23–1.6) $\times10^{-7}$ eV Å^{-1}，最大误差达到 (0.23–1.8) $\times10^{-6}$ eV Å^{-1}，分别比 fp32 下限低 9–55 倍和 9–38 倍。在 fp32 下，JET-MLIPs 内核 + 表与仅 JET-MLIPs 内核的差异，在力分量上至多为 (6.9–10) $\times10^{-6}$ eV Å^{-1}，在能量上至多为 (0.7–2.1) $\times10^{-8}$ eV 原子$^{-1}$，位于$N=2048$ （主图 3c）。对于 $LiCoO_2$ （360 个原子）、金刚石硅（216 个原子）和非晶硅 O_2 （560 个原子）各自的十个分子动力学帧，JET-MLIPs 的两种模式的均方根误差均保持在 fp32 e3nn 均方根误差的 0.93–1.07 倍以内，最大误差均不超过其 1.6 倍；而孤立 fp64 表误差比 fp32 下限低 43–188 倍（表 S4）。

确定性。JET-MLIPs 卷积的前向传播在累加时不使用浮点原子操作，并且在全部五个系列中均具有确定性；而反向传播中的原子加法会导致力不确定。在这十个 5632 原子电解质帧上，重复评估 JET-MLIPs 内核 + 表时，任一力分量的差异最大为 (1.7–2.4)$\times10^{-6}$ eV Å^{-1}；若将仅核变体纳入比较，差异最高可达 2.6×10^{-6} eV Å^{-1}。相应地，fp32 e3nn（7net-nano-5.5、7net-nano-6.0 和 7net-0）的差异为 (3.1–5.9) $\times10^{-6}$ eV Å^{-1}，FlashTP（全部四个模型）的差异为 (3.6–6.4) $\times10^{-6}$ eV Å^{-1}。重复运行 e3nn 和 FlashTP 评估时，总能量也可能相差一个最后一位单位。在其余十一个模型中，JET-MLIPs 内核+ 表同样表现出比现有 fp32 TP 加速后端更小的运行间力差异（(1.1–5.5) $\times10^{-6}$ eV Å^{-1}，相比之下为 (1.4–21) $\times10^{-6}$ eV Å^{-1}；见表 S3）。残余能量变化

最高达 13 个最后一位单位的 MACE 能量变化源于 MACE 内部对原子能量进行求和时图形处理器 `scatter_add` 的非确定性，而非卷积所致。总体而言，JET-MLIPs 在全部 15 个模型中均满足主文第 2.4 节所述的单次调用精度预算，唯一例外是 NequIP-OAM L 的能量偏移。

S3.2 分子动力学验证

运行间发散。对于 7net-nano-5.5，我们从 5632 原子电解质在 300开尔文下达到平衡的同一状态（位置和速度）出发，分别使用上游 FlashTP 运行了三条 10 皮秒的微正则系综轨迹，并使用 JET-MLIPs 内核 + table 运行了三条（表S5b）。运行采用 0.5 飞秒时间步长，不使用 D3；每 10 步保存一次热力学输出，每 100 步保存一次位置。混合后端配对与相同后端配对的发散速率相同，均方根偏差衰减时间为 109–115 飞秒。在 2–10 皮秒期间（分为八个 1 皮秒区块），FlashTP 的温度范围为 302.57–303.13 开尔文，JET-MLIPs 的温度范围为 302.67–303.25 开尔文，标准误差为 0.12–0.26 开尔文。在同一时间段内，势能相差不超过 8.9×10^{-5} eV 原子$^{-1}$，压力范围为 12,521–12,703 bar，能量漂移保持在 1.0×10^{-4} meV 原子$^{-1}$ 皮秒$^{-1}$以内。

100皮秒和2纳秒内的能量守恒。对于7net-nano-5.5和7net-0，每个模型均在完整的5632原子晶胞上进行能量最小化，并在300开尔文下进行10皮秒的NVT系综模拟，以此初始化。随后，我们使用上游 FlashTP、仅 JET-MLIPs 内核和 JET-MLIPs 内核 + 表（表S5a和图S4），在不使用D3的情况下以0.5 飞秒步长运行100皮秒的微正则系综模拟。每次运行的能量漂移均不超过 3×10^{-5} 毫电子伏特 原子$^{-1}$ 皮秒$^{-1}$，即100皮秒内小于 3×10^{-6} 电子伏特 原子$^{-1}$，且没有系统性的偏向。将7net-nano-5.5的运行扩展至2纳秒，并采用两种JET-MLIPs模式（图S5），整个过程未发生原子损失；由于总能量恒定下结构持续弛豫，温度同样升高了3–4开尔文。2纳秒内，总能量漂移为 1.7×10^{-7}（内核 + 表）和 -1.7×10^{-6} 毫电子伏特 原子$^{-1}$ 皮秒$^{-1}$（仅内核）。后者累计变化量为3.4微电子伏特 原子$^{-1}$，与线性拟合附近2.7微电子伏特 原子$^{-1}$ 的均方根波动相当。

热力学和结构性质。在100 ps运行的后半段（表S5a）中，三种后端在势能方面的差异小于 1.5×10^{-4} eV atom^{-1}，温度差异为1.1开尔文，压力差异为0.4%。最大差异反映了总能量恒定时势能与动能之间的交换（$2\Delta E_{\mathrm{pot}}/3k_B \approx 1.2$开尔文）。从相同状态重复运行 JET-MLIPs 内核 + 制表，结果最多相差 5×10^{-5} eV atom^{-1}、0.4开尔文和0.9%的压力。仅内核与内核 + 制表的差异最多为 6×10^{-5} eV atom^{-1}、0.5开尔文和0.4%，表明制表偏差未超过运行间发散。第一个锂–氧和锂–氟径向分布函数峰的位置相差不超过一个 bin（0.025埃），积分至2.7埃所得的配位数差异不超过0.06（图S4c,d,g,h）。较大的正压力（12500和8400 bar）是由于在以7net-0加D3达到平衡的体积下省略了D3。100皮秒内呈次线性增长的Li均方位移（图S4b,f）主要由单次跳跃主导，因此不用于衡量等效性。

其他系列的NVE漂移。 我们还在完整的5632原子晶胞上测量了其他系列的NVE能量漂移（第 S1.1节）。每条轨迹均采用以下方式进行平衡：使用三个初始速度种子之一，在 10 皮秒 的NVT系综下，以 300开尔文（Nosé–Hoover 恒温器， 0.05 皮秒 阻尼）进行平衡。随后，在NVE积分前，使用固定种子将速度重新初始化为 300开尔文 ，并始终采用 0.5 飞秒 的时间步长。NVE 运行的平均温度为 295–306 开尔文。

在 10 皮秒的运行中，MACE-MP-0 medium 使用 JET-MLIPs 内核 + 表时的漂移为 -1.4×10^{-4} 至 2.7×10^{-4} 毫电子伏/原子/皮秒，而使用 cuEquiv-ariance 时为 -1.6×10^{-4} 至 2.4×10^{-4} 毫电子伏/原子/皮秒。NequIP-OAM M 使用 JET-MLIPs

内核 + 表，漂移为 -3.7×10^{-5} 至 1.9×10^{-4} 毫电子伏/原子$^{-1}$ 皮秒$^{-1}$，使用 OpenEquivariance。这些 10 皮秒运行的种子间范围为 1.2×10^{-4} 至 4.1×10^{-4} 毫电子伏/原子$^{-1}$ 皮秒$^{-1}$，范围较宽，决定了比较的分辨率。该分辨率约比 100 皮秒 SevenNet 运行（第 S3.2 节）的界限粗十倍。

使用 JET-MLIPs 内核 + 表格，nequix-mp-1 和 EquFlash V1 采用三个随机种子进行的 10 皮秒运行，其漂移分别为 -1.2×10^{-4} 至 1.2×10^{-4} 和 -1.4×10^{-4} 至 6.3×10^{-6} meV 原子$^{-1}$ 皮秒$^{-1}$。MACE-MP-0 medium、NequIP-OAM M 和 nequix-mp-1 采用一个随机种子进行的 100皮秒运行，其漂移分别为 -7.1×10^{-5}、-1.5×10^{-5} 和 -1.2×10^{-5} meV 原子$^{-1}$ 皮秒$^{-1}$。每次运行中，总能量的标准差均为 (2.3–3.7) $\times 10^{-3}$ meV 原子$^{-1}$。

S4 性能、内存与分子动力学集成

本节为第 3.2–3.5 节的性能结果提供支持。内容涵盖其他 SevenNet 后端和邻居列表重建、JET-D3、步骤时间线、模型调用尺度扩展、分子动力学集成、更大的晶胞和致密固体、内存行为以及消费级 GPU 基准测试。运行协议和内存定义见第 2.6 节和 S1.3 节。

S4.1 其他 SevenNet 后端与邻居列表重建 s

主表2 比较了每个模型与其上游模型及现有 TP 加速后端的表现，图S9c 展示了其他 SevenNet 后端。未加速的 e3nn 后端在图形处理器上运行，但不使用专用 TP 内核；FlashTP 则采用上游 SevenNet 的 pair style。cuEquivariance 和 OpenEquivariance 后端使用基于 Kokkos 的 ML-IAP、上游 SevenNet 0.13.0 和上游 D3；下文将这一组合称为上游 ML-IAP 部署。JET-MLIPs 的两种模式均通过 `e3gnn` 运行，或通过 JET-MLIPs ML-IAP 包装器运行（第S2.5节，图S9c 中的空心柱）。

与不含 D3 的上游 FlashTP 相比，JET-MLIPs 内核 + 表快 6.3–12。1× 倍；而仅通过 `e3gnn` 运行的 JET-MLIPs 内核快 1.9–4。6× 倍，涵盖四个 SevenNet 模型（图S9c）。在上游后端中，除 7net-nano-5.5 外，cuEquivariance 速度最快；在该模型上，OpenEquivariance 略快（19.16 ms，而 cuEquivariance 为 19.94 ms）。在主表2 和图S9c 的重复运行中，SevenNet 无重建步耗时的最大值与最小值之差，至多为中位数的 3.4%。例外情况是 cuEquivariance 在 7net-nano-5.5 上的差值为 10%（运行结果集中在 18.1 ms 和 20.0 ms 左右），在 7net-nano-6.0 上为 5.2%。对于另外四个系列，JET-MLIPs 的差值至多为 0.9%，现有 TP 加速后端至多为 0.6%；例外是 NequIP-OAM S 的现有 TP 加速后端，差值为 7.2%。

邻居列表重建。 上游 cuEquivariance 的重建开销（第 3.2 节）体现在包含重建的平均步耗时中。在不使用 D3 的情况下，将电解质在 450 个计时步数中的 9–10 次重建计入后，其平均步耗时分别从 19.94、21.11、45.30 和 187.9 毫秒/步$^{-1}$ 增加至 33.33、36.35、62.27 和 397.3 毫秒/步$^{-1}$，对应于 7net-nano-5.5、7net-nano-6.0、7net-0 和 7net-omni。图 S9c 中的其他后端仍与不含重建时的数值相差不超过 0.7 毫秒。MACE 现有的 TP 加速后端使用相同的 cuEquivariance 库，却没有出现此类开销，因为对于四个 MACE 模型，其包含重建的平均步耗时与不含重建的步耗时相差仍不超过 0.6 毫秒。鉴于这一对比，该开销似乎是上游 SevenNet 集成所特有的；若将重建计入，表 2 中报告的加速比还会进一步提高。

这一惩罚会随系统规模增大。在规模扫描中，每种规模均采用首次运行的数据，每次重建成本为 $(t_{\mathrm{all}} - t_{\mathrm{pure}}) \times 450/N_{\mathrm{rebuild}}$，其中 t_{all} 是包含重建的平均步耗时，t_{pure}是无重建步耗时，N_{rebuild} 是在 450 个计时步数中的重建次数。在 5632 个原子时，7net-nano-5.5、7net-nano-6.0、7net-0 和 7net-omni 的每次重建成本分别为 0.61、0.75、0.86 和 9.5 秒。在 45,056 个原子时，7net-nano-5.5、7net-0 和 7net-nano-6.0 的成本分别升至 4.0、4.5 和 4.7 秒。对于 7net-omni，在 11,264 和 22,528 个原子时，成本分别升至 16.1 和 27.5 秒。固体扫描不涉及邻居列表重建。

S4.2 JET-D3，aGPU色散校正

JET-D3 (`pair_style jet/d3`) 在 JET-MLIPs 中以 LAMMPS 成对样式实现 D3(BJ) 色散校正，并防止机器学习原子间势加速后，色散计算成为分子动力学步的主要耗时。

它独立于模型系列，不过下文所有 D3 计时均使用 SevenNet 模型；上游 D3 指的是 SevenNet 的 `pair_style d3`。

方法。 原子对按 32×32 分块处理，每个分块使用一个线程束，并且每个无序对只访问一次。在每个周期方向上的长度均至少为 1.5 倍截断值的模拟晶胞中，原子会被排序到空间分箱中，并跳过原子对不可能位于截断半径内的分块。与配位数相关的 C_6 插值按原子和伙伴类型进行因式分解，使每对原子约需 10 次融合乘加，而不是使用带除法的 5×5 参考循环。缓冲区使用普通设备内存分配，而上游代码则为每个原子使用统一内存。D3 在独立的 CUDA 流上执行。与 `e3gnn`结合时，它会在边构建之前启动，从而实现并发执行，使色散计算不会与卷积串行执行。

与上游 D3 的一致性。 JET-D3 与上游 D3 一致，适用于 5632、11、264 和 22、528个原子，误差在 $|\Delta E|/N \le 1.4\times10^{-9}\,\mathrm{eV\,atom^{-1}}$、$|\Delta P| \le 8.9\times10^{-4}\,\mathrm{bar}$ 和 $|\Delta F| \le 1.2\times10^{-7}\,\mathrm{eV\,Å^{-1}}$范围内。力差比单精度浮点数的力分布范围低一个数量级。两种实现与双精度参考 s-dftd3 的差异均为 $4.4 \times 10^{-8}\,\mathrm{eV Å^{-1}}$ 均方根（第 S3.1节）。

含 D3 的步时间。纳入 D3 后，JET-MLIPs 内核相较于上游 SevenNet 后端的加速比 + 表明，其针对 cuEquivariance 的加速比分别达到 22.4、21.0、12.5 和 4。$8\times$ 针对 FlashTP 的加速比分别达到 26.2、26.1、14.9 和 8。$8\times$ 以上数据依次对应 7net-nano-5.5、7net-nano-6.0、7net-0 和 7net-omni（图S9d）。JET-D3 仅增加 2.6–2。7 毫秒的步时间$^{-1}$，而上游 D3 则使 cuEquivariance 增加 119–130 毫秒的步时间$^{-1}$，使 FlashTP 增加 123–186 毫秒的步时间$^{-1}$。因此，上游 FlashTP 使用 D3 时，色散计算占其 7net-nano-5.5 步时间的 76%。

独立尺度扩展。 在针对电解质副本的独立 D3 基准测试中（每种尺寸首次运行，D3 设置见第 S1.1 节），JET-D3 在 18.8 毫秒/步$^{-1}$ 下处理 5632、11,264 和 22,528 个原子。上游 D3 的步时间为 92.2 、 145 和 926 毫秒 / 步 $^{-1}$，对应的加速比分别为 29 、 20 和 $49\times$。在这三种尺寸下，JET-D3 的设备内存几乎保持不变，为 533–545 MiB；而上游 D3 则需要 783、1295 和 3471 MiB。在 45,056 个原子时，上游 D3 因 `cudaErrorInvalidAddressSpace`而终止，而 JET-D3 则可扩展至 360,448 个原子，步时间为499 毫秒$^{-1}$，内存占用为 797 MiB。

S4.3 步骤时间线与卷积块

图 S6a、b 展示了使用 D3 时最后 60 个微正则系综步骤中的 50 个步骤的 Nsight Systems 时间线。对于 7net-nano-5.5 和 7net-0，上游 FlashTP 每步分别占用 52 和 91 ms 的图形处理器内核时间，其中 D3 分别占 31 和 30 ms，TP 内核分别占 12.5 和 44 ms。主机每步发起 500 次和 825 次启动，其余部分对应的墙钟步时间分别为 149.5 和 197.7 ms。这些墙钟时间是五次运行中包含重建的平均步耗时的中位数，离散范围为 13–15%；而相同运行中不含重建的中位数分别为 162.2 和 203.6 ms。使用 JET-MLIPs 时（图 S6c、d），主机每步发起一次 CUDA 图启动和四次内核启动，使该步骤受图形处理器限制，空闲时间为 0.8–1.0 ms。仅内核模式下的图形处理器内核时间分别为 10.4 和 32.0 ms，使用该表时分别为 4.9 和 12.4 ms，其中使用该表时卷积分别耗时 1.9 和 7.3 ms。D3 占 1.6–1.9 ms，图形处理器边构建占 0.06 ms，其余 290 和 530 个 PyTorch 与 cuBLAS 内核分别占 1.3 和 3.5 ms。制表后，

因此，7net-nano-5.5 中除 D3 外的其余模块（1.4 ms）所耗时间接近其卷积时间（1.9 ms）的四分之三。

图 S9a、b 报告了卷积块的执行时间和峰值分配。e3nn、FlashTP、cuEquivariance 和 OpenEquivariance 卷积块在预计算的边嵌入上运行；边嵌入的计算在每次模型调用中需要 1.0–2.6 毫秒，并未计入其中。JET-MLIPs 卷积块在内核中计算边几何（使用 `split` 仅内核模式）。对于 e3nn、FlashTP、cuEquivariance 和 OpenEquivariance，当边嵌入仅计算一次时，卷积块在 7net-nano-5.5 和 7net-0 的一次模型调用中占 73–98%（调用时间见图 S7）。对于 JET-MLIPs，该比例降至 67–85%（仅内核）和 36–53%（内核+ 制表）。制表后，其余模块因此占模型调用的 47–64%，这与第 3.3 节所述的转变一致。仅将 e3nn 张量积替换为 FlashTP，就能将 7net-nano-5.5 的卷积时间从 82.8 毫秒降低至 18.7 毫秒。因此，尽管未融合张量积和聚合在算术运算中所占份额较小，却至少占据了该时间的四分之三（第 2.2 节），表明其成本主要由内存流量而非算术运算决定。

对于 MACE，使用 $N = 4096$在 H200 上对 5632 原子电解质进行的 Nsight Systems 性能分析显示，对于 MACE-MP-0 medium，每步的融合卷积占 GPU 内核时间的 65%（9.7 毫秒中的 6.4 毫秒）；对于 MACE-MP-0 large，则占 44%（13.0 毫秒中的 5.7 毫秒）。这些性能分析中启用了第 2.5 节所述的按物种分组的自连接内核。

S4.4 模型调用扩展

第 3.5 节中的模型调用扫描通过不断复制 5632 原子晶胞，直至各后端的 GPU 内存耗尽，从而确定其余 SevenNet 后端的容量上限（图 6a–d）。对于 7net-nano-5.5，FlashTP 和 OpenEquivariance 可容纳 180,224 个原子，e3nn 可容纳 45,056 个原子。对于 7net-0，FlashTP 和 OpenEquivariance 可容纳 90,112 个原子，与 cuEquivariance 相当；e3nn 则可容纳 22,528 个原子。对于 7net-omni，张量积加速器可容纳 22,528 个原子，而 e3nn 在 5632 个原子时会超出 GPU 内存。这些容量均低于 JET-MLIPs 内核在 + 第 3.5节表格中达到的容量。

在同一扫描的模型调用计时中，JET-MLIPs 内核 + 表在所有四种 SevenNet 模型的各个规模下都实现了最短的模型调用时间，包括截断半径较大的 7net-nano-6.0（图S7）。在 5632 个原子时，7net-nano-5.5、7net-0 和 7net-omni 分别需要 6.4、14.9 和 67.7 毫秒。仅 JET-MLIPs 内核的性能始终优于 FlashTP，但在 7net-0（32.1 毫秒，对比 30.0 毫秒）和 7net-omni（215 毫秒，对比 148 毫秒）上慢于 cuEquivariance。与仅 JET-MLIPs 内核相比，在 5632 个原子时，该表将调用时间缩短了 3。2× （7net-omni）、2。2× （7net-0）和 1。7× （7net-nano-5.5），这就是第3.3节中所述的范围。

S4.5 分子动力学集成：消融与步的份额

消融。 在消融研究中，我们分别禁用了图重放、GPU边构建和 D3 重叠，并在 `e3gnn` 中启用 D3（图 S9g，三次运行的中位数）。基准测试使用仅用于计时的输入，时间步长为 0.1 飞秒，并采用 `fix nve/limit 0.01` 以抑制邻居列表重建。GPU边构建带来的加速最大，因为禁用该功能后，7net-nano-5.5 表格模式的步时间会从 6.24 ms 增至 11.07ms。对于三个较小的模型，图重放使表格模式的步时间缩短 2.7–8.7%，仅内核模式缩短 2.6–3.5%；D3 重叠则使这三个较小模型在表格模式下的步时间缩短 1.7–6.8%。在 SevenNet 之外，在 JET-MLIPs ML-IAP 封装器中启用图重放后，352 原子的 EquFlash V1 步时间缩短 20%（从 74.0 ms 降至 59.0 ms/步$^{-1}$），整卡GPU内存也从 4155 MiB 降至 3347 MiB。

可归因于内核的加速。 为将内核性能与分子动力学集成区分开来，我们在上游 SevenNet ML-IAP 部署中仅执行 JET-MLIPs 内核（第 S4.1节）。对于 7net-nano-5.5、7net-nano-6.0、7net-0 和 7net-omni，所需步时间分别为 14.04、15.99、44.93 和 243.5 ms/步$^{-1}$，对应于 cuEquivariance 速度的 1.42、1.32、1.01 和 0.77 倍，以及 OpenEquivariance 速度的 1.36、1.41、1.12 和 1.34 倍。对于 7net-omni，该部署相较于 `e3gnn` (209.5 ms/步$^{-1}$) 产生 34.0 ms 的开销；据 Nsight Systems 分析，其中 26.1 ms（77%）来自负责打包和解包幽灵原子数据的 Kokkos 内核。

上游步骤的小晶胞下限。 在 360 个原子（钴酸锂$_2$）和 5632 个原子（电解质）下对 7net-nano-5.5 进行性能分析，可以分离出主文第 3.4 节小晶胞下限背后的两项固定成本。首先，上游 ML-IAP 部署在每一层都交换幽灵原子特征，且不进行图重放。相较于 `e3gnn`，在 360 个原子时，这种交换使每步增加 5.4–6.9 毫秒；在 5632 个原子时增加 5.4–5.5 毫秒，其中约 3.7 毫秒来自 60 个 Kokkos 幽灵交换内核。其次，即时执行的 cuEquivariance 模型受主机限制。在 360 个原子时，其 13.7–13.8 毫秒的步时间中，图形处理器仅占用 6.1 毫秒，而主机发起了 587 个模型内核；对于两个纳米模型，直到 1152 个原子时，步时间仍接近 14 毫秒。通过 `e3gnn`JET-MLIPs 内核 + 表可以消除这两项开销，并在两个规模下分别实现 1.25 和 3。49 毫秒的步时间$^{-1}$。JET-MLIPs ML-IAP 包装器同样通过将幽灵原子折叠到其所属原子中，并将调用作为单个图进行重放来避免这些开销；由于主机同步和 ML-IAP 输入准备，剩余开销为 0.4–1。1 毫秒的步时间 $^{-1}$，高于 `e3gnn`。

NequIP-OAM 图形处理器时间和包装器控制。 在 5632 原子电解质的 Nsight Systems 跟踪中（160 步中的最后 60 步），JET-MLIPs 内核 + 表将图形处理器占用时间从 OpenEquivariance 的 5.47 降至 1。60 毫秒 （NequIP-OAM S（3。4×）），并将 M（3。8×）的时间从 43.4 降至 11。4 毫秒。包含径向多层感知机的稠密矩阵乘积，分别占 OpenEquivariance 为 S 和 M 配置测得的图形处理器时间的 25%（1.36 毫秒）和 36%（15.6 毫秒）。为将内核性能与分子动力学集成隔离开来，还在相同的 JET- MLIPs CUDA 图包装器中评估了 OpenEquivariance；对于 NequIP-OAM S、M 和 L，其每步耗时分别为 3.8、37.7 和 125。7 毫秒$^{-1}$，而 JET-MLIPs 内核 +表的耗时分别为 2.0、11.7 和 37。9 毫秒$^{-1}$（三次运行中无需重建的中位数，相同测试框架列，主表 2）。

S4.6 体系尺寸与致密固体

为测试不同材料之间的尺度扩展（正文第 3.4 节和正文图 5），我们比较了电解质、钴酸锂$_2$, 金刚石硅和非晶 SiO_2。它们的原子密度分别为 0.097、0.124、0.050 和0.070 原子 埃$^{-3}$。扫描使用了第 S1.1 节中的晶胞，并采用首次运行（第 S1.3节）的无需重建计时结果。SevenNet 在这些固体中的静态精度见表 S4。

幂律拟合$\log t = \alpha \log N + c$（图 S8）对满足以下条件的点采用普通最小二乘法拟合：$N \geq 4000$ 个原子，且至少需要三个这样的点。钴酸锂$_2$, 只有两个点，阈值以下的其他扫描也无法进行拟合。根据残差和 $n-2$自由度估算标准误差，且不计运行间的变化。纳入原子数少于 4000 的晶胞后，由于固定的每步开销， α降至 0.28–1.00。相比之下，两个最大规模之间的斜率趋近于线性（例如，在硅中使用 cuEquivariance 的 7net-nano-5.5，其斜率为 0.95，规模介于125，000和 262，144个原子之间）。

S4.7 内存：现有 TP 加速后端的反向变体与限制

表 S6 比较了 SevenNet 的仅内核反向变体（`recompute` 和默认值`split`，第 S2.2 节）与 JET-MLIPs 内核 + 表。相较于 `recompute`， `split` 使 7net-omni 的分子动力学步耗时降低 7.5%，使 7net-0 降低 3.8%，而对两个纳米模型没有可测量的影响。其每边 256 B 的缓冲区使 7net-nano-5.5 在 5632 个原子时的模型调用内存增加 36%，在 360,448 个原子时增加 55%（12.2 吉比字节，相比 7.9 吉比字节；正文图 6a）。对于 7net-nano-6.0，该缓冲区使可容纳的最大系统规模从 2,883,584 个原子减半至 1,441,792 个原子。该表既不需要逐边 MLP 存储，也不需要 MLP 重新计算。它仅增加表本身（每个模型 17–227 MiB）和一个临时的谐波梯度缓冲区（第 S2.3 节）。它可处理与 `recompute` 相同的最大系统，且分子动力学运行速度比 `split`快 2.5–3.3× 倍。

在 SevenNet LAMMPS 扫描中，cuEquivariance 在两个纳米模型达到 360,448 个原子、7net-0 达到 152,064 个原子以及 7net-omni 达到 45,056 个原子时发生内存不足故障。其可支持的最大 LAMMPS 尺寸（152,064、45,056 和 22,528 个原子）均不超过模型调用扫描的结果（180,224、90,112 和 22,528 个原子，见第 S4.4 节），这与幽灵原子和 ML-IAP缓冲区所需的额外内存相符。对于其他四个系列，表 S7 列出了内存不足限制，以及正文第 3.5 节中 4–55% 范围背后的整块 GPU 内存占比。在 5632 原子晶胞中，EquFlash V1 的现有 TP 加速后端已占用 H200 内存的 94%（整块 GPU 峰值，未列于表 S7）。

S4.8 消费级 GPU

在 WSL2 下的 12 GB RTX 4070 SUPER 上，我们在 5632 原子电解质上对 SevenNet（图 S9e）、MACE 和 NequIP-OAM（图 S9f）进行了基准测试，而未在该图形处理器上评估 Nequix 和 EquFlash。报告的步耗时为包含重建操作的平均值，取三次运行的中位数（NequIP-OAM 为七次运行）。

SevenNet 的基准测试构成了正文第 3.5 节中消费级 GPU 吞吐率的依据。当整块 GPU 使用量超过 12,282 MiB 中的 11,800 MiB，且尚未因 WSL2 溢出到主机内存时，该运行被归类为内存不足。两种 JET-MLIPs模式均可在测试的最大体系（45,056 原子）上运行 7net-nano-5.5 和 7net-0，并可使 7net-omni 运行至 11,264 原子，但在 22,528 原子时失败。在上游后端中，只有 FlashTP 能够在 5632 原子下运行 7net-0，而 FlashTP 和 cuEquivariance 均无法运行 7net-omni。OpenEquivariance 在运行 7net-0 时就已超出内存限制，因此未对 7net-omni 进行评估。所有上游后端均可在 11,264 原子下运行 7net-nano-5.5，并在 22,528 原子时失败；而对于 7net-0，FlashTP 在 11,264 原子时失败。使用 JET-D3 时，5632 原子内核 + 表格步所需时间为 13.62、34.92 和 240.5 ms 步$^{-1}$，分别对应 7net-nano-5.5、7net-0 和 7net-omni；与不含 D3 的运行相比，步耗时增加了 6.7– 6.9 ms 步$^{-1}$。由于在 WSL2 下其每步需要数秒，而在 H200 上仅需 0.12–0.19 s，因此未在该图形处理器上对上游 `pair_style d3` 进行基准测试。

对于 MACE 和 NequIP-OAM，两个后端均通过 ML-IAP 执行，其中 $N = 4096$ 适用于 MACE。现有的 TP 加速后端无法容纳三个 MACE 模型或 NequIP-OAM L，而对于 NequIP-OAM S 和 M，JET-MLIPs 内核 + 表的速度分别快 3.3 和 3.8× 倍。在 352 原子晶胞上，JET-MLIPs 内核 + 表能够将 NequIP-OAM 现有单精度浮点数后端的力复现到 (1.15–1.30) $\times 10^{-5}$ eVÅ^{-1} 以内。JET-MLIPs 内核 + 表的全程峰值 PyTorch 分配量，对于 MACE-MP-0 medium、MACE-MPA-0 medium 和 MACE-MP-0 large 分别为 735、3094 和 1480 MiB；对于 NequIP-OAM S、M 和 L 分别为 202、656 和 1413 MiB。相比之下，OpenEquivariance 在 NequIP-OAM S 和 M 上分别为 677 和 6015 MiB。NequIP-OAM M 在默认分配器下测得的 OpenEquivariance 步时间，并且

降至 1　　21.9 毫秒/步$^{-1}$（PyTorch 分配器上限设为 GPU 内存的 80% 时）。

S5 应用示例

本节介绍正文第 3.6 节所述两种输运应用的模拟与分析细节，以及正文表 3 和表 4。实验值仅用于比较，既未用于拟合，也未用于选择输入。

碳酸酯液体的剪切黏度。三个模拟晶胞分别包含纯 EMC（670 个分子，10,050 个原子）、由 65 个 $LiPF_6$ 溶于 292 个 EC 和 440 个 EMC 中组成的溶液（10,040 个原子），以及由 148 个 $LiPF_6$ 溶于相同溶剂中组成的溶液（10,704 个原子）。溶剂的 EC:EMC 体积比为 3:7，对应的 EC 质量分数为 36 wt%，而用于比较的实测 1 M 电解质中该数值为 30 wt% [16]。在 NPT 平均体积下，盐浓度分别为 0.94 和 2.02 mol L^{-1}，密度分别为 1.179 和 1.280 g cm^{-3}。模拟采用 7net-nano-6.0 加 D3（e3gnn，内核 + 表，JET-D3），在 298.15 开尔文下以 1 飞秒时间步长进行。每种液体先在 NPT系综中平衡 1 ns，随后在三个 NVT系综生产重复样本中各采样 10 ns。

剪切黏度通过格林–久保积分计算，积分基于每 2 飞秒采样一次的五个剪切应力分量，并在每个重复样本的六个区块上取平均，每个区块为 1667 皮秒。纯 EMC、0.94 mol L^{-1} 溶液和 2.02 mol L^{-1} 溶液的黏度值分别在 50 皮秒、100 皮秒和 150 皮秒的积分截断时间下提取，不确定性以三个重复样本的标准误差表示。2.02 mol L^{-1} 晶胞中的所有 148 个 PF_6^- 阴离子在生产阶段均保持完整。在对 0.94 mol L^{-1} 晶胞进行的 50 皮秒 NVE 验证中，时间步长为 1 飞秒并启用 D3（平均温度为 300.7 开尔文），能量漂移保持在 2.8×10^{-4} meV 原子$^{-1}$ 皮秒$^{-1}$。

Li_6PS_5Cl 中的锂扩散分析。对于正文第 3.6 节中的锂扩散系数，均方位移根据每 1 皮秒保存一次的展开锂坐标计算，并将每一帧用作时间原点。计算均方位移之前，先减去锂质心的位移。当达到 14 埃^2 所需的滞后时间不超过运行时长的四分之一时，才接受该轨迹。相对标准偏差按照 He 等人（正文中引用）的定义计算，为 $3.43/\sqrt{N_{\text{eff}}}+0.04$。此处，$N_{\text{eff}}$ 为拟合窗口内的最大均方位移乘以锂原子数，再除以 a^2；其中 a 为 Li 24g 位点的最近邻距离（3.13–3.14 埃）。在十条轨迹中（$T = 300$–400 开尔文，$x = 0.25$–0.615），固定生产晶胞的晶格参数为 9.751–9.779 埃，生产运行时长为 4.5–20.0 纳秒。均方位移拟合窗口在 400 开尔文时为 161–2622 皮秒，在 300 开尔文时为 3796–10000 皮秒；相对标准偏差为 5.2–6.7%。在 400 开尔文和 $x = 0.615$ 条件下进行的一致性检查中，使用 FlashTP，以相同的模型、JET-D3、初始结构和分析方法重复运行。其平均 NPT 系综体积为 201,592 埃^3,，而 JET-MLIPs 的平均体积为 201,604 埃^3 。

应用单元计时和内存。 图中展示了两个应用单元的步耗时。S9d，测量时使用了恒体积生产输入（1 飞秒 用于电解质，2 飞秒 用于 Li_6PS_5Cl），结果为三次运行的中位数。使用 e3nn、FlashTP 和 JET-MLIPs 内核 +表时，电解质的峰值 GPU 内存分别为 82.9、12.8 和 1.3 吉比字节；Li_6PS_5Cl 的峰值 GPU 内存分别为 37.4、5.4 和 1.2 吉比字节。

S6 支持的布局和限制

这些内核支持 $l_{\max} \leq 4$ 以及每层通道重数为 8 的倍数、最高达 128 的路径与通道张量积，其中包括 7net-omni 中某一次数的两种奇偶性。更宽的层需要

适配器拆分（将 256 通道的 MACE-MP-0 large 拆为两个 128 通道的部分，并将 256 通道的 EquFlash 拆为 32 通道的切片）。仅内核模式支持 SevenNet 的贝塞尔(8)基组和包含 64 个单元的两个隐藏层，而表格内核接受边长和元素对的径向函数。执行仅限于预编译的层形状。五个系列发布的形状均已针对构建所用图形处理器的共享内存和寄存器限制预先编译（第 S2.3节），支持新形状则需要重新构建库。图重放仅限于单个 MPI 秩，且未对多GPU执行进行基准测试。

对于 MACE-MPA-0 和 MACE-OMAT-0，其径向变换取决于元素对，表格内核会为所含元素的每个无序对存储一张表。在两个层中，这些表每对占用 112 MiB，分别对应 $N = 4096$ $(4N_W$ 个单精度浮点值/节点， $N_W = 512$ 和 1280）。电解质中的六种元素构成 21 个无序对，共占用 2352 MiB。这是这些模型全程峰值为 3.07 GiB、而 MACE-MP-0 medium 为 0.76 GiB 的差异的主要原因；后者只需一组表（正文表 2）。该内存用量随所含元素数 $s(s+1)/2$ 而变化。 s

补充表

表 S1：已部署模型的每条边算术运算量。三个 SevenNet 检查点逐层列出输入不可约表示（不可约表示）、门控保留的有效张量积（TP）路径、径向多层感知机输出宽度N_W，以及径向多层感知机和 TP 的每条边融合乘加（FMA）计数。作为对比，还列出了 MACE-MP-0 medium 的所有卷积路径（$N_B = 10$，三个含 64 个单元的隐藏层）和 NequIP-OAM M 的所有卷积路径（$N_B = 8$，一个含 128 个单元的隐藏层）。径向多层感知机的计数为 $N_B H + (n_h - 1)H^2 + HN_W$（正文第 2.2节），TP 的计数为 $\sum_p f_p\, m_{l_1(p)}$，其中 f_p 是路径 的非零克莱布施–戈尔丹系数数量（称为纤维，图 S1b、c），每个系数在每个通道上需要一次融合乘加。因此，当一条路径平均少于 H 个纤维时，径向多层感知机的计算量占主导。不可约表示记作 $m \times lp$，表示具有 m 个通道、次数为 l 且奇偶性为 p（e 表示偶，o 表示奇）。SO(3) 模型 7net-0 将偶奇偶性（e）赋予所有不含反演宇称的不可约表示。模型整体属性汇总于表 S2。

模型	层	输入不可约表示	路径	N_W	多层感知机融合乘加	张量积 融合乘加
7net-nano-5.5 [17]	L0	$32 \times 0e$	3	96	10,752	288
	L1	$32 \times (0e + 1o + 2e)$	11	352	27,136	2656
	L2	$32 \times (0e + 1o + 2e)$	3	96	10,752	288
7net-0 [18]	L0	$128 \times 0e$	3	384	29,184	1152
	L1–L3	$128 \times 0e + 64 \times 1e + 32 \times 2e$	15	960	66,048	6848
	L4	$128 \times 0e + 64 \times 1e + 32 \times 2e$	3	224	18,944	480
7net-omni [12]	L0	$128 \times 0e$	4	512	37,376	2048
	L1	$128 \times 0e + 64 \times 1o + 32 \times 2e + 32 \times 3o$	34	1760	117,248	24,864
	L2	$128 \times 0e + 64 \times (1o + 1e) + 32 \times (2o + 2e + 3o + 3e)$	64	3008	197,120	47,680
	L3	$128 \times (0o + 0e) + 64 \times (1o + 1e) + 32 \times (2o + 2e + 3o + 3e)$	34 个，共 68 个，	1760	117,248	24,864
	L4	$128 \times 0e + 64 \times 1o + 32 \times 2e + 32 \times 3o$	4	256	20,992	704
MACE-MP-0 中型	L0	$128 \times 0e$	4	512	41,600	2048
	L1	$128 \times (0e + 1o)$	10	1280	90,752	11,008
NequIP-OAM M	L0	$32 \times 0e$	3	96	13,312	288
	L1–L2	$128 \times 0e + 64 \times 1o + 32 \times 2e$	11	768	99,328	4416
	L3	$128 \times 0e + 64 \times 1o + 32 \times 2e$	3	224	29,696	480

表 S2：已发布的 15 个模型的架构规格和规模组。总参数数量和径向 MLP 参数占比（径向 MLP 中参数所占的百分比）是针对已发布的未转换检查点而言的，且 $l_{\max}$ 表示卷积中的最高球谐阶数。每个原子的边数是在包含 5632 个原子的电解质晶胞上计算的。模型按参数数量分为小型（$< 10^6$）、中型（10^6–10^7）和大型（$> 10^7$）类别。MACE-OMAT-0 在 OMat24 [19]上训练。

组	模型	参数 (M)	径向 MLP（%）	层	$l_{\max}$	截断半径 (Å)	边 每个原子的
小型	7net-nano-5.5	0.10	55.0	3	2	5.5	68.1
	7net-nano-6.0	0.10	55.0	3	2	6.0	88.2
	NequIP-OAM S	0.62	5.6	2	1	4.5	35.5
	nequix-mp-1	0.71	31.9	4	3	6.0	88.2
	nequix-oam-1	0.71	31.9	4	3	6.0	88.2
	7net-0	0.84	29.2	5	2	5.0	50.2
中型	NequIP-OAM M	3.19	7.6	4	2	6.0	88.2
	MACE-MP-0 中型	4.69	2.8	2	3	6.0	88.2
	MACE-MPA-0 中型	9.06	1.5	2	3	6.0	88.2
	MACE-OMAT-0 中型	9.06	1.5	2	3	6.0	88.2
	NequIP-OAM L	9.61	7.7	6	3	6.0	88.2
大型	MACE-MP-0 大型	15.8	1.6	2	3	4.5	35.5
	7net-omni	25.8	2.3	5	3	6.0	88.2
	EquFlash V1	28.7	11.1	10	3	6.0	88.2
	EquFlash V2	44.9	6.4	10	4	6.0	88.2

表 S3：相对于 fp64 的单点力和能量误差，以及重复调用时的力差异。 在一台 H200 上，对十个电解质分子动力学帧进行评估时，各 fp32 后端相对于同一模型 fp64 参考值的力分量误差（eV $Å^{-1}$）和每原子能量均方根误差（eV $atom^{-1}$）（5632 个原子；对于 7net-omni、NequIP-OAM L 和 EquFlash，则为 352 个原子）。对于 SevenNet，fp64 参考值为 fp64 e3nn；对于其他系列，则为各系列发布的 fp64 模型。对于非 SevenNet 系列，所有评估（包括 fp64 参考值）在以其原生精度（fp32 或 fp64）构建边向量时，均使用经过 fp32 舍入且完全相同的位置和晶胞矢量。因此，最大力误差仅能在各系列内部进行比较，而均方根误差受此影响较小。JET-MLIPs 内核 + 表对 MACE 使用 $N = 4096$，对其他所有系列使用 $N = 2048$，并额外列出一个位于 $N = 2048$的 MACE-MP-0 medium 条目。SevenNet 评估返回 fp32 总能量，而 MACE 的 fp32 原子能量求和会同等影响所有后端。对于 EquFlash，力精度作为主要验证指标，且 fp64 参考值以 fp64 累加原子能量。最后两列报告了同一帧上两次连续 fp32 评估之间的最大和均方根力分量差异（eV $Å^{-1}$）。对于 SevenNet，所报告的最大值涵盖 5632 个原子时十个帧中四个模型的全部结果，而重复调用的均方根差异未予记录。

		相对于双精度浮点数的误差			重复调用，$\lvert\Delta F\rvert$	
模型	后端	均方根 $\lvert\Delta F\rvert$	最大值 $\lvert\Delta F\rvert$	均方根 $\lvert\Delta E\rvert/N$	max	rms
7net-nano-5.5	e3nn	1.42×10^{-6}	1.56×10^{-5}	1.52×10^{-7}	5.48×10^{-6}	–
	FlashTP	1.42×10^{-6}	1.44×10^{-5}	1.26×10^{-7}	6.44×10^{-6}	–
	仅 JET-MLIPs 内核	1.36×10^{-6}	1.30×10^{-5}	1.73×10^{-7}	2.15×10^{-6}	–
	JET-MLIPs 内核 + 表	1.42×10^{-6}	1.28×10^{-5}	1.26×10^{-7}	2.38×10^{-6}	–
7net-nano-6.0	e3nn	1.51×10^{-6}	1.51×10^{-5}	9.26×10^{-8}	5.90×10^{-6}	–
	FlashTP	1.51×10^{-6}	1.46×10^{-5}	1.20×10^{-7}	6.20×10^{-6}	–
	仅 JET-MLIPs 内核	1.47×10^{-6}	1.12×10^{-5}	1.72×10^{-7}	2.62×10^{-6}	–
	JET-MLIPs 内核 + 表	1.48×10^{-6}	1.30×10^{-5}	1.04×10^{-7}	2.38×10^{-6}	–
7net-0	e3nn	1.29×10^{-6}	1.21×10^{-5}	1.68×10^{-7}	3.13×10^{-6}	–
	FlashTP	1.29×10^{-6}	1.21×10^{-5}	1.82×10^{-7}	3.58×10^{-6}	–
	仅 JET-MLIPs 内核	1.18×10^{-6}	1.12×10^{-5}	2.08×10^{-7}	1.91×10^{-6}	–
	JET-MLIPs 内核 + 表	1.27×10^{-6}	1.25×10^{-5}	1.74×10^{-7}	1.67×10^{-6}	–
7net-omni (352)	e3nn	1.27×10^{-6}	8.72×10^{-6}	3.29×10^{-7}	–	–
	FlashTP	1.27×10^{-6}	8.63×10^{-6}	3.42×10^{-7}	4.17×10^{-6}	–
	仅 JET-MLIPs 内核	1.24×10^{-6}	8.70×10^{-6}	2.72×10^{-7}	2.15×10^{-6}	–
	JET-MLIPs 内核 + 表	1.27×10^{-6}	7.61×10^{-6}	3.43×10^{-7}	1.79×10^{-6}	–
MACE-MP-0 中型	现有 TP 加速单精度浮点数	4.54×10^{-6}	1.14×10^{-4}	2.73×10^{-4}	4.0×10^{-6}	4.7×10^{-7}
	JET-MLIPs 内核 + 表	4.54×10^{-6}	1.10×10^{-4}	2.72×10^{-4}	1.5×10^{-6}	2.3×10^{-7}
	JET-MLIPs 内核 + 表，2048 个节点	5.50×10^{-6}	1.28×10^{-4}	2.67×10^{-4}	–	–
MACE-MP-0 large	现有的 TP 加速单精度浮点数	4.82×10^{-6}	1.14×10^{-4}	1.45×10^{-4}	3.6×10^{-6}	5.4×10^{-7}
	JET-MLIPs 内核 + 表	4.67×10^{-6}	1.13×10^{-4}	1.46×10^{-4}	1.2×10^{-6}	2.0×10^{-7}
MACE-MPA-0 中型	现有 TP 加速单精度浮点数	6.77×10^{-6}	1.12×10^{-4}	2.87×10^{-5}	7.2×10^{-6}	1.3×10^{-6}
	JET-MLIPs 内核 + 表	4.80×10^{-6}	1.13×10^{-4}	2.66×10^{-5}	2.4×10^{-6}	3.4×10^{-7}
MACE-OMAT-0中型现有 TP 加速单精度浮点数		8.96×10^{-6}	1.12×10^{-4}	3.24×10^{-5}	1.8×10^{-5}	2.1×10^{-6}
	JET-MLIPs 内核 + 表	4.86×10^{-6}	1.06×10^{-4}	2.99×10^{-5}	3.3×10^{-6}	4.7×10^{-7}
NequIP-OAM S	现有 TP 加速单精度浮点数	7.70×10^{-6}	1.06×10^{-4}	3.21×10^{-7}	2.1×10^{-5}	2.0×10^{-6}
	JET-MLIPs 内核 + 表	5.33×10^{-6}	1.05×10^{-4}	2.35×10^{-7}	5.5×10^{-6}	6.7×10^{-7}
NequIP-OAM M	现有的 TP 加速单精度浮点数	5.54×10^{-6}	1.08×10^{-4}	5.51×10^{-7}	5.5×10^{-6}	7.3×10^{-7}
	JET-MLIPs 内核 + 表	5.38×10^{-6}	1.15×10^{-4}	2.77×10^{-7}	3.6×10^{-6}	5.2×10^{-7}
NequIP-OAM L	现有的 TP 加速单精度浮点数	3.50×10^{-6}	2.68×10^{-5}	1.78×10^{-7}	5.4×10^{-6}	1.1×10^{-6}
	JET-MLIPs 内核 + 表	3.58×10^{-6}	2.86×10^{-5}	6.05×10^{-7}	2.5×10^{-6}	5.1×10^{-7}
nequix-mp-1	现有的 TP 加速单精度浮点数	4.64×10^{-6}	1.11×10^{-4}	1.42×10^{-7}	1.9×10^{-6}	3.0×10^{-7}
	JET-MLIPs 内核 + 表	4.59×10^{-6}	1.13×10^{-4}	1.11×10^{-7}	1.5×10^{-6}	2.2×10^{-7}
nequix-oam-1	现有的 TP 加速单精度浮点数	4.63×10^{-6}	1.10×10^{-4}	2.45×10^{-7}	2.8×10^{-6}	3.8×10^{-7}
	JET-MLIPs 内核 + 表	4.57×10^{-6}	1.09×10^{-4}	2.45×10^{-7}	2.1×10^{-6}	2.8×10^{-7}
EquFlash V1	现有的 TP 加速单精度浮点数	1.77×10^{-6}	2.55×10^{-5}	1.65×10^{-6}	1.4×10^{-6}	3.0×10^{-7}
	JET-MLIPs 内核 + 表	1.81×10^{-6}	2.24×10^{-5}	1.30×10^{-6}	1.2×10^{-6}	2.1×10^{-7}
EquFlash V2	现有的 TP 加速单精度浮点数	参考（双精度浮点数不适用）			2.3×10^{-6}	3.7×10^{-7}
	JET-MLIPs 内核 + 表	1.49×10^{-6}	1.13×10^{-5}	1.49×10^{-6}	1.1×10^{-6}	2.1×10^{-7}

表 S4：致密固体中相对于 fp64 的单点精度。 在列标题所示单元中，fp32 e3nn、仅 JET-MLIPs 内核（k.o.）和 JET-MLIPs 内核 + 表（k.+t.）相对于 fp64 e3nn 进行评估，指标包括均方根和最大力分量误差（eV $Å^{-1}$）以及每原子能量均方根误差（eV $atom^{-1}$）。孤立 fp64 表的均方根力误差是相对于 fp64 径向多层感知机计算的，其比值表示 fp32 e3nn 均方根力误差除以该孤立表误差（$N = 2048$；每个体系和模型使用十个帧，在一台 H200 上进行）。构型从 300 K NVT 轨迹中每隔 0.5 皮秒采样一次，使用 0.5 飞秒时间步长，通过 JET-MLIPs 内核+ 表进行计算，并在超胞复制后重新初始化速度。括号表示 $LiCoO_2$, 金刚石硅（8 原子立方晶胞的$3 \times 3 \times 3$超胞）和非晶硅 O_2（70 原子晶胞的$2 \times 2 \times 2$超胞）的原子数。

		均方根 $\lvert\Delta F\rvert$ (10^{-7})			最大值 $\lvert\Delta F\rvert$ (10^{-6})			均方根 $\lvert\Delta E\rvert/N$ (10^{-7})		单独的表，双精度浮点数	
体系	模型	e3nn	k.o.	k.+t.	e3nn	k.o.	k.+t.	e3nn	k.o.	k.+t.	均方根（10^{-8}）	比率
钴酸锂$_2$(360)	7net-nano-5.5	6.47	6.73	6.04	3.59	3.63	2.91	3.17	2.46	2.97	1.40	46
	7net-0	5.71	5.61	5.94	3.49	3.10	3.17	2.60	2.69	2.73	0.784	73
	7net-omni	7.60	7.50	7.20	3.92	3.86	3.39	3.39	3.07	2.95	0.693	110
硅 (216)	7net-nano-5.5	8.81	8.45	8.19	3.10	3.14	2.87	2.12	1.70	1.95	0.934	94
	7net-0	7.25	7.17	7.20	3.53	2.57	3.28	2.01	2.17	2.17	0.385	188
	7net-omni	8.05	7.53	8.38	3.24	3.09	3.13	2.46	2.66	2.66	0.765	105
非晶硅氧化物$_2$ (560)	7net-nano-5.5	17.9	17.7	19.2	9.67	14.6	10.9	3.74	3.74	4.21	4.17	43
	7net-0	15.7	15.7	16.5	9.23	10.1	8.85	3.69	3.69	3.69	2.04	77
	7net-omni	18.1	18.4	17.7	10.3	13.2	9.47	5.49	6.13	4.88	2.22	82

表 S5：热力学平均值与 NVE 轨迹发散。（a）使用上游 FlashTP、仅 JET-MLIPs 内核和 JET-MLIPs 内核进行的 100皮秒 NVE 运行中，后半段（50–100 皮秒）的平均值 + 表，其中 PE 表示势能。（b）从相同状态初始化的多组 10 皮秒 7net-nano-5.5 NVE 运行之间的轨迹发散，其中 JET-MLIPs 表示内核 + 表（完整的 5632 原子电解质、一个 H200、0。5 飞秒 时间步长，无 D3）。RMSD 是轨迹之间原子位置差的均方根，而力的均方根大小为 1。41 eV 埃$^{-1}$。 $|\Delta T|$ 各列报告超过所示阈值的首次通过时间。（b）中的第0步力差异在 LAMMPS 分子动力学运行中计算，因此与第 S3.1 节中独立模型调用的离散度略有不同。

(a) 热力学平均值和能量漂移

模型	后端	漂移（毫电子伏 原子$^{-1}$ 皮秒$^{-1}$）	T（开尔文）	势能（电子伏 原子$^{-1}$）	P（巴）
7net-nano-5.5	FlashTP（上游）	-6.8×10^{-6}	305.8	-5.74572	12,477
	仅 JET-MLIPs 内核	-2.2×10^{-5}	304.7	-5.74557	12，452
	JET-MLIPs 内核 + 表	-1.5×10^{-5}	304.9	-5.74560	12,494
7net-0	FlashTP（上游）	1.4×10^{-5}	300.9	-5.74597	8375
	仅 JET-MLIPs 内核	1.4×10^{-5}	301.5	-5.74606	8373
	JET-MLIPs 内核 + 表	1.2×10^{-5}	301.1	-5.74600	8348

(b) 重复运行的发散

	步-0 $\|\Delta F\|$ (eV 埃$^{-1}$) 首次		$\|\Delta T\|$ 以上（皮秒）		RMSD	RMSD	
运行对	max	rms	10^{-3} 开尔文	1 K	倍增时间（飞秒）	1 皮秒（毫埃）	10 皮秒（埃）
FlashTP 与 FlashTP（3 对）	6.7×10^{-6}	1.2×10^{-6}	0.74–0.79	1.77–1.89	110	0.9–1.1	2.07–2.09
JET-MLIPs 与 JET-MLIPs（3 对）	2.4×10^{-6}	4.9×10^{-7}	0.68–0.74	1.82–1.95	109	1.0–1.4	2.02–2.08
FlashTP 与 JET-MLIPs（9 对）	1.8×10^{-5}	2.8×10^{-6}	0.68–0.76	1.60–1.85	115	1.0–2.0	1.98–2.09

表 S6：Seven- Net 中仅内核反向变体与径向制表的比较。各列分别报告无需重建的 LAMMPS 步时间、单次模型调用的 PyTorch 峰值分配（吉比字节）以及可容纳的最大系统规模（原子数）（步时间和峰值分配采用 5632 原子电解质；单个 H200、微正则系综、不使用 D3）。JET-MLIPs 仅内核使用 `recompute` 或默认的 `split` 变体；JET-MLIPs 内核 + 表使用默认的 N。

模型	变体	毫秒/步$^{-1}$	峰值（吉比字节）	最大运行规模（原子数）
7net-nano-5.5	仅内核，`recompute`	8.60	0.187	2,883,584
	仅内核，`split`	8.62	0.254	2，883，584
	内核 + 表	**3.49**	0.205	2,883,584
7net-nano-6.0	仅内核，`recompute`	10.68	0.196	2，883，584
	仅内核，`split`	10.69	0.292	1,441,792
	内核 + 表	**4.09**	0.213	2,883,584
7net-0	仅内核，`recompute`	31.51	0.312	1,441,792
	仅内核，`split`	30.31	0.327	1，441，792
	内核 + 表	**10.94**	0.419	1,441,792
7net-omni	仅内核，`recompute`	226.4	1.665	360，448
	仅内核，`split`	209.5	1.784	360，448
	内核 + 表	**63.43**	1.918	360,448

表 S7：现有 TP 加速后端在系统规模上限时的 GPU 内存消耗。在一块 H200 上（nvidia-smi, 总容量为 143，771 MiB），比较了现有 TP 加速后端与 JET-MLIPs 内核 + 表在相同系统规模下占用的整块 GPU 内存比例。列出的原子数对应现有后端首次出现 CUDA 内存不足（OOM）的情况；但在钴酸锂$_2$ 和非晶硅O_2, 中，NequIP-OAM M 和 nequix-mp-1 完成了扫描范围内的最大规模。现有的 TP 加速后端包括：用于 MACE 的 cuEquivariance、用于 NequIP-OAM 的 OpenEquivariance、用于 nequix-mp-1 的搭载 OpenEquivariance 内核的 PyTorch 后端，以及用于 EquFlash V1 的 GGNN。此比较未纳入 EquFlash V2。

			现有 TP 加速后端	JET-MLIPs 内核 + 表	
模型	体系	原子	占比（%）	MiB	占比（%）
MACE-MP-0 中等电解质		45,056	OOM	6353	4.4
	钴酸锂 2	39,936	OOM	6043	4.2
	Si	64,000	OOM	8539	5.9
	非晶硅氧化物 2	70,000	OOM	8675	6.0
NequIP-OAM M	电解质 152,064		OOM 20，957		14.6
	钴酸锂 2	39,936	96	6193	4.3
	Si	262,144	OOM	31,459	21.9
	非晶硅氧化物 2	70,000	99	9397	6.5
nequix-mp-1	电解质 152，064		OOM	34，051	23.7
	钴酸锂 2	39,936	80	9997	7.0
	Si	125,000	OOM	28,309	19.7
	非晶硅氧化物 2	70,000	84 16	,779	11.7
EquFlash V1	电解质	11，264	OOM	22，817	15.9
	钴酸锂 2	39,936	OOM 79	,593	55.4
	Si	17,576	OOM	33,891	23.6
	非晶硅氧化物 2	15,120	OOM 29	,747	20.7

补充图

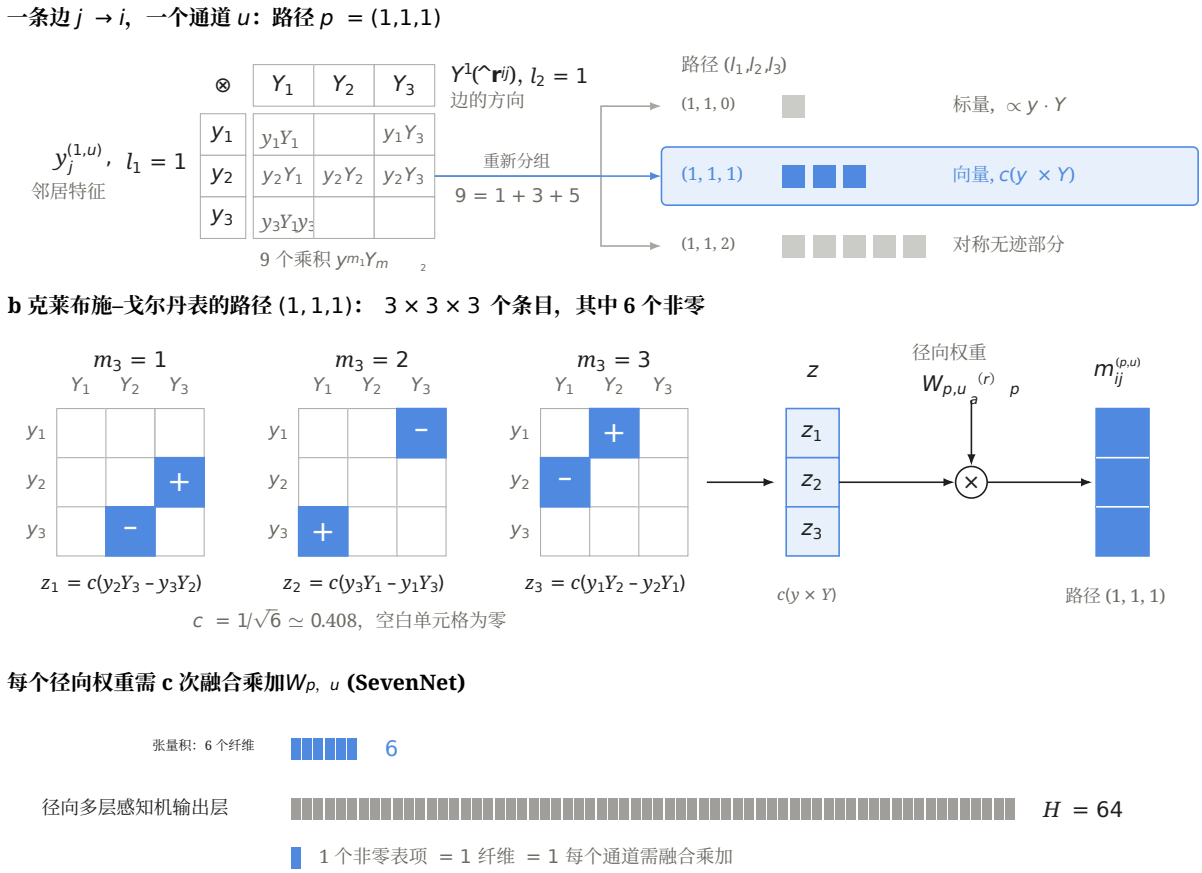


图 S1：作为稀疏系数表的张量积路径。a，对于一条边 $j \to i$ 和一个通道 u，1 阶邻居特征 $y_j^{(1,u)}$ 与边谐波 $Y^1(\hat{\mathbf{r}}_{ij})$ 产生九个乘积$y_{m_1} Y_{m_2}$。这些乘积重新分组为 0、1 和 2 阶，分别包含一个、三个和五个分量（每个输出阶数 l_3 对应一条路径）；其中路径 $p = (l_1, l_2, l_3) = (1, 1, 1)$ 保留了蓝色高亮分支。**b**，这条路径的克莱布施–戈尔丹表为每个输出分量 z_{m_3} 包含一个 3×3 切片，行 y_{m_1} 和列 Y_{m_2} 按照 e3nn 的分量顺序排列。每个单元格存储其乘积的系数。27 个单元格中有六个非零，分别包含 $+c$ 或 $-c$，并在固定路径归一化 a_p 之前带有 $c = 1/\sqrt{6} \simeq 0.408$。径向权重 $W_{p,u}(r)$ 将 $z = c(y \times Y)$ 的三个分量缩放为边消息 $m_{ij}^{(p,u)}$。**c**，每个非零单元格对应一个纤维，每个通道需要一次融合乘加（FMA）。因此，每个径向权重对应的路径需要六次 FMA；相比之下，SevenNet 径向多层感知机的输出层需要 $H = 64$ 次 FMA 才能生成该权重。

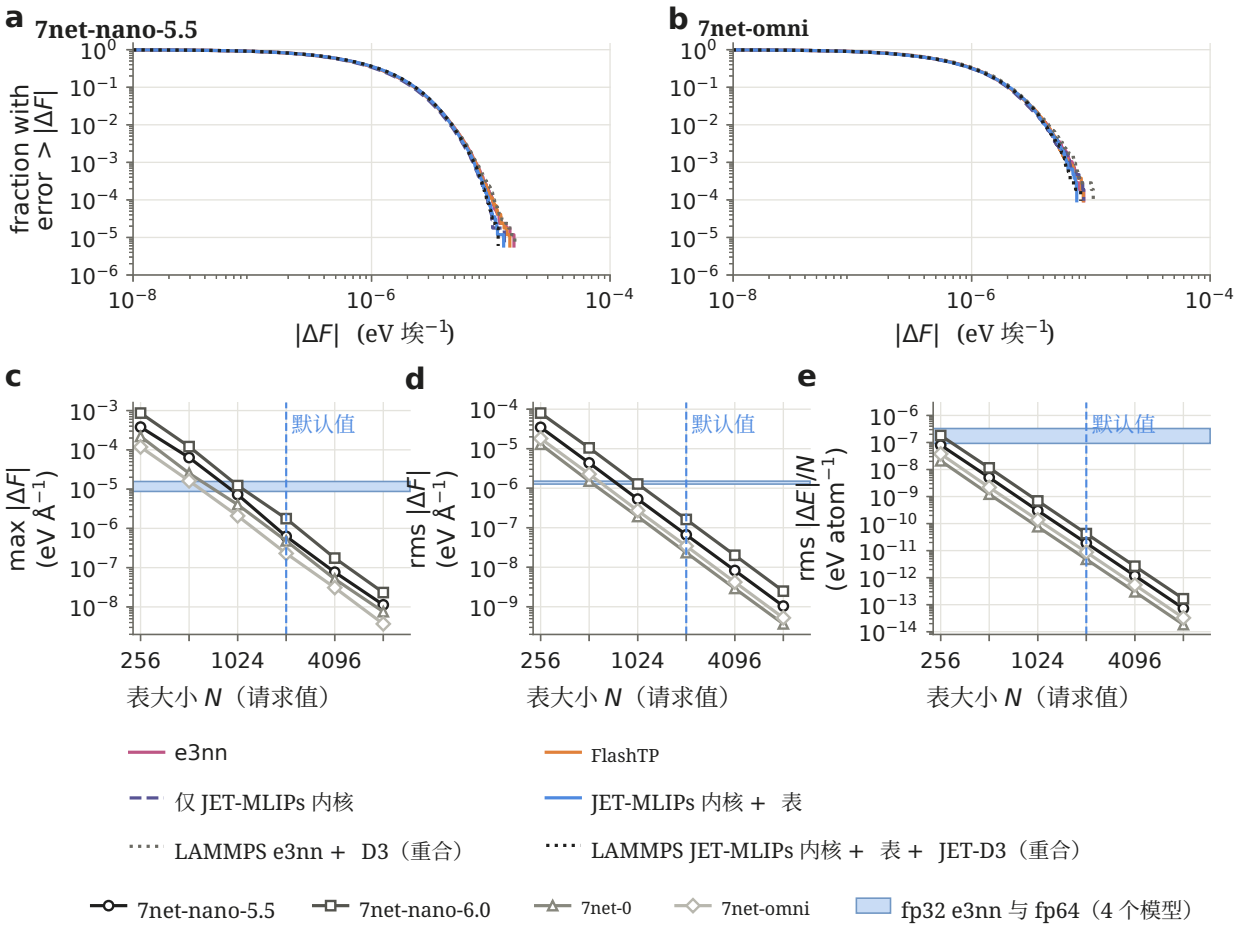


图 S2：相对于 fp64 的力误差分布。(a,b) 7net-nano-5.5 和 7net-omni 的力分量中，超过 $|\Delta F|$ 的比例（使用 10 个电解质分子动力学帧，分别含有 5632 和 352 个原子，使用一块 H200 ）。D3 部署使用 fp64 e3nn + s-dftd3 作为参考。(c–e) 在对齐网格上，fp64 表相对于 fp64 径向多层感知机的最大力误差和均方根力误差（eV 埃$^{-1}$），以及每原子能量均方根误差（eV 原子$^{-1}$），随请求节点数 N 的变化；使用相同的帧。浅蓝色带表示四个模型相对于 fp64 的 fp32 e3nn 误差范围，虚线表示默认值N。模型和后端编码的定义见图例。

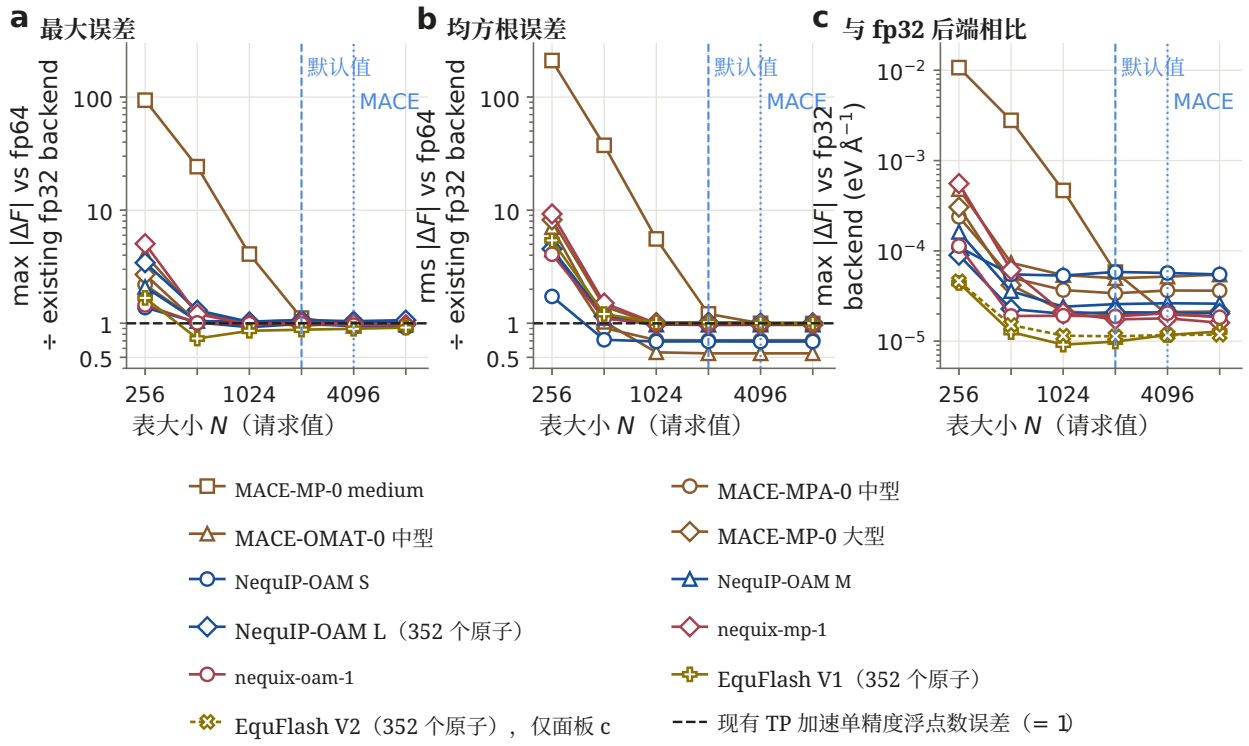


图 S3：其他系列中的径向表精度。 (a,b) 表大小为 N 时，fp32 JET-MLIPs 内核 + 表相对于 fp64 的最大力误差和均方根力误差，除以现有 TP 加速后端的 fp32 力误差（5632 原子电解质；NequIP-OAM L 和 EquFlash 为 352 个原子；一块 H200）。(c) fp32 JET-MLIPs 内核 + 表与同一模型现有 fp32 TP 加速后端之间的最大力差。图中仅在 (c) 中显示 EquFlash V2（虚线），因为其 fp64 参考值超出 GPU 内存容量。垂直虚线和点线分别标示默认值 2048 和 MACE 的 4096。

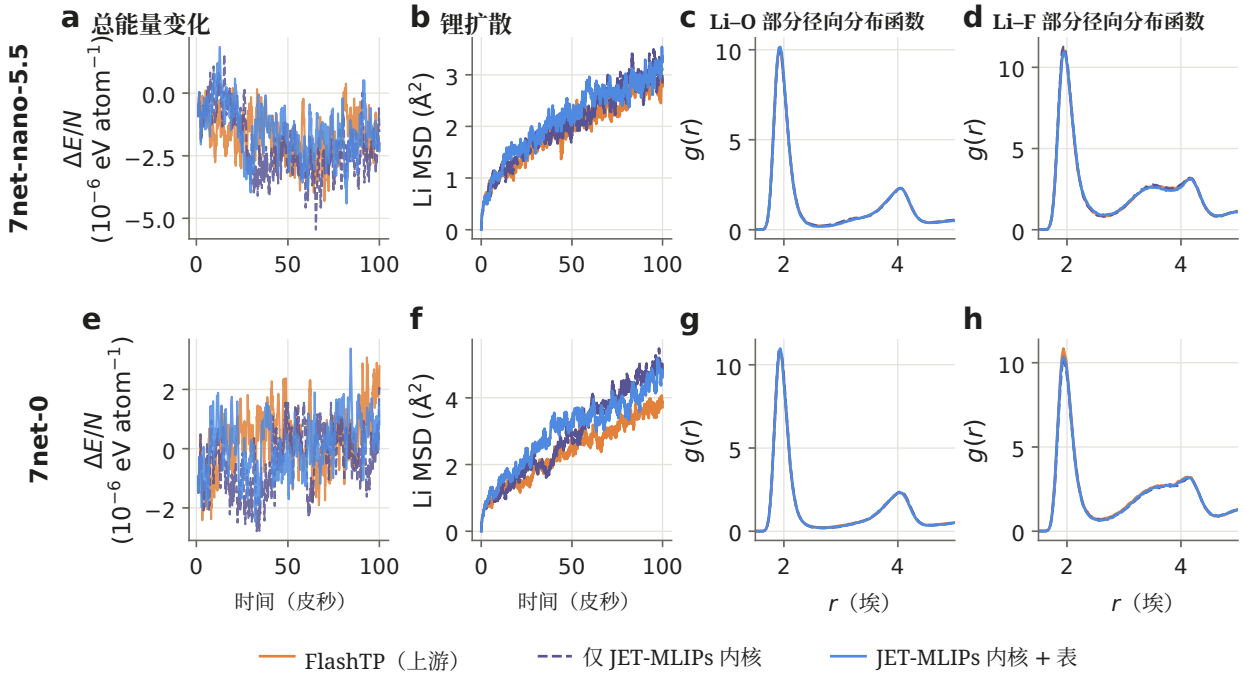


图 S4：NVE 分子动力学中的能量守恒和锂配位。(a–d) 7net-nano-5.5。(e–h) 7net-0（完整的 5632 原子电解质，单个 H200，100皮秒 NVE，0.5 飞秒步长，不使用 D3）。各列分别显示相对于初始值的每原子总能量变化（10^{-6} eV atom^{-1}，1皮秒滑动平均）、锂均方位移（MSD，Å^2），以及随距离（Å）变化的无量纲锂–氧和锂–氟部分径向分布函数（RDF）。后端编码的定义见图例。

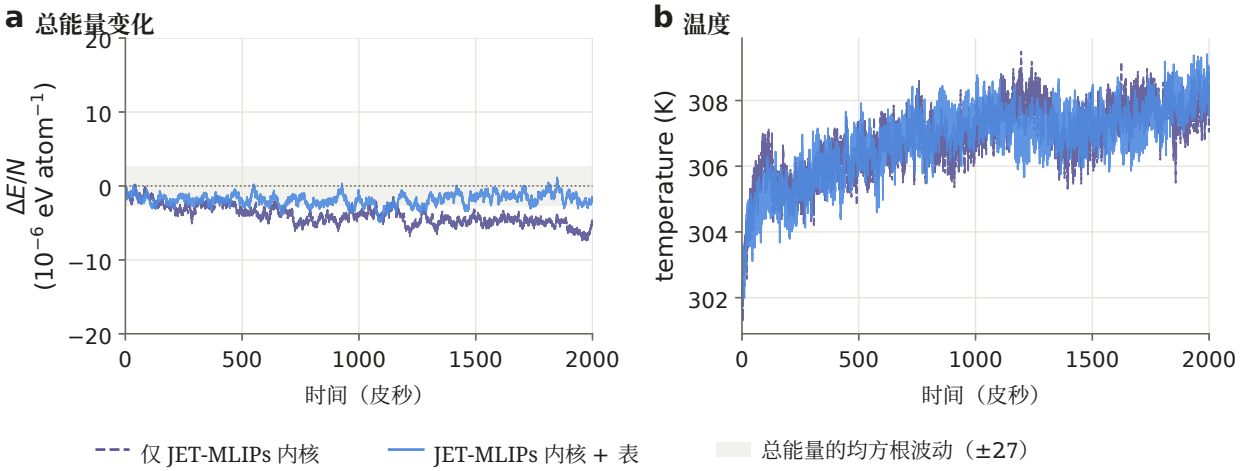


图 S5：使用 7net-nano-5.5 进行两纳秒 NVE 模拟。(a) 相对于初始值的每原子总能量变化（10^{-6} eV atom^{-1}）。灰色区域表示零附近的 $\pm 2.7 \times 10^{-6}$ eV atom^{-1}，即未经平滑的总能量相对于其线性拟合的均方根波动。(b) 温度。两者均为 1皮秒滑动平均（完整的 5632 原子电解质，单个 H200，2纳秒 NVE，0.5 飞秒步长，不使用 D3；两个后端使用相同的初始位置和速度）。

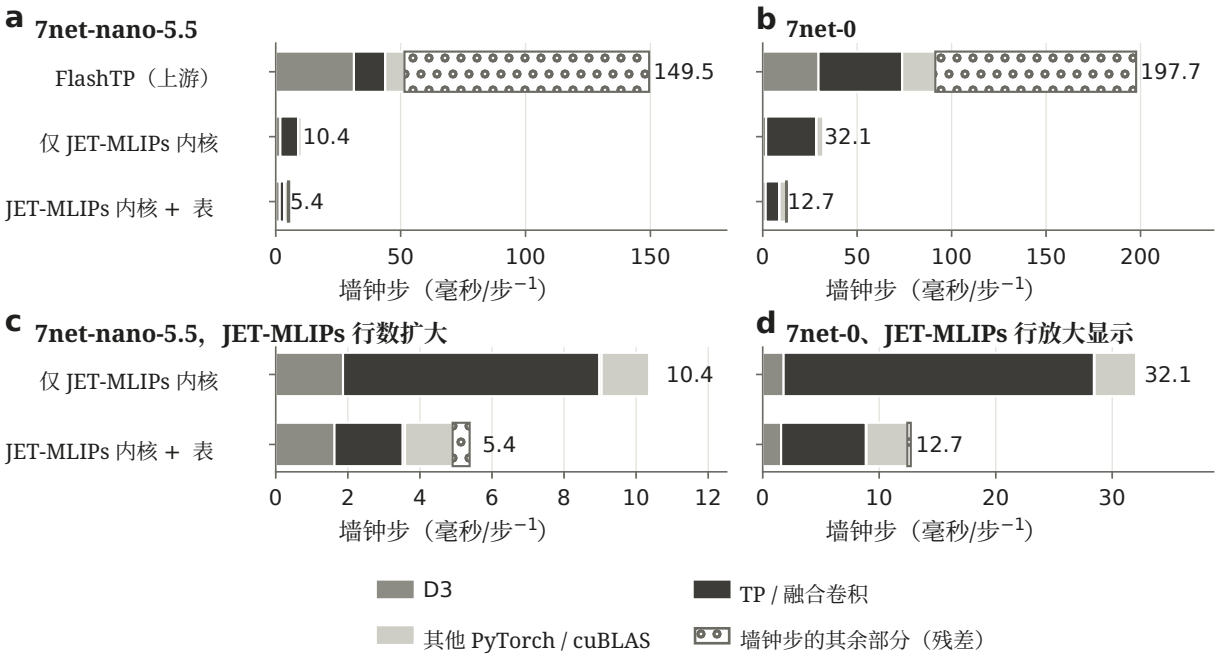


图 S6：一个分子动力学步的图形处理器时间线。（a、b）7net-nano-5.5 和 7net-0 的图形处理器内核时间分类及墙钟时间残差（5632 原子电解质、一个 H200、使用 D3 的微正则系综、Nsight Syste ms、60 步中的最后 50 步）。（c、d）JET-MLIPs 行的放大视图。残差是从包含重建的未分析平均步时间中减去已分析内核时间总和所得，并包含主机开销，而不是测得的空闲时间。条形末端标签表示墙钟时间，图例定义了各类别。对于 7net-0 的仅 JET-MLIPs 内核（其内核时间总和等于步时间），或低于 0.3 ms 的情况不绘制残差，其中包括 0.1 ms 的 7net-nano-5.5 仅内核。由于上游墙钟时间在不同运行之间有所变化，上游残差为近似值（第 S4.3 节）。0.06 ms 步$^{-1}$ 图形处理器边构建在（c）中以灰色细线表示。

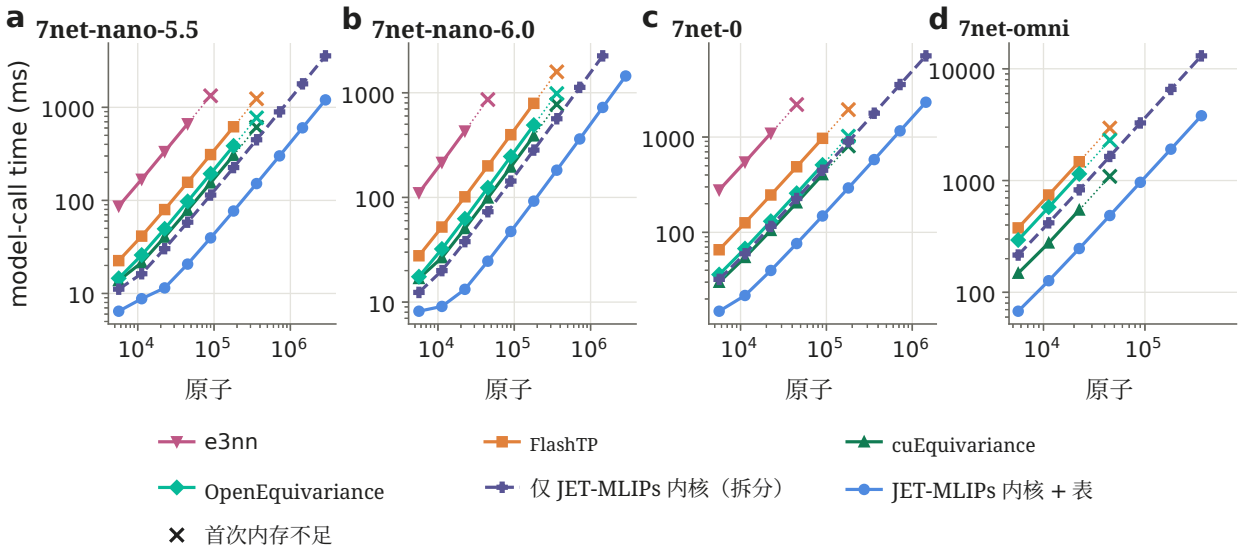


图 S7：模型调用时间与系统规模的关系。不同原子数下，前向计算加力计算的平均 CUDA 事件时间（毫秒）（5632 原子电解质副本，单个 H200，每种构型的首次基准测试运行）。(a–d) 各面板中标出了 SevenNet 模型；仅 JET-MLIPs 内核使用默认值 `split`，JET-MLIPs 内核 + 表使用默认值 N。十字标记表示首次超出内存容量的系统规模，其时间由最后一次拟合结果线性外推得出，并以无测量数据的细点线连接。(d) 中，e3nn 在 5632 个原子时就已超出 GPU 内存，因此没有曲线。后端标记见图例，其中仅内核以虚线表示，其他构型以实线表示。计时针对独立模型调用，不包括 LAMMPS 集成。

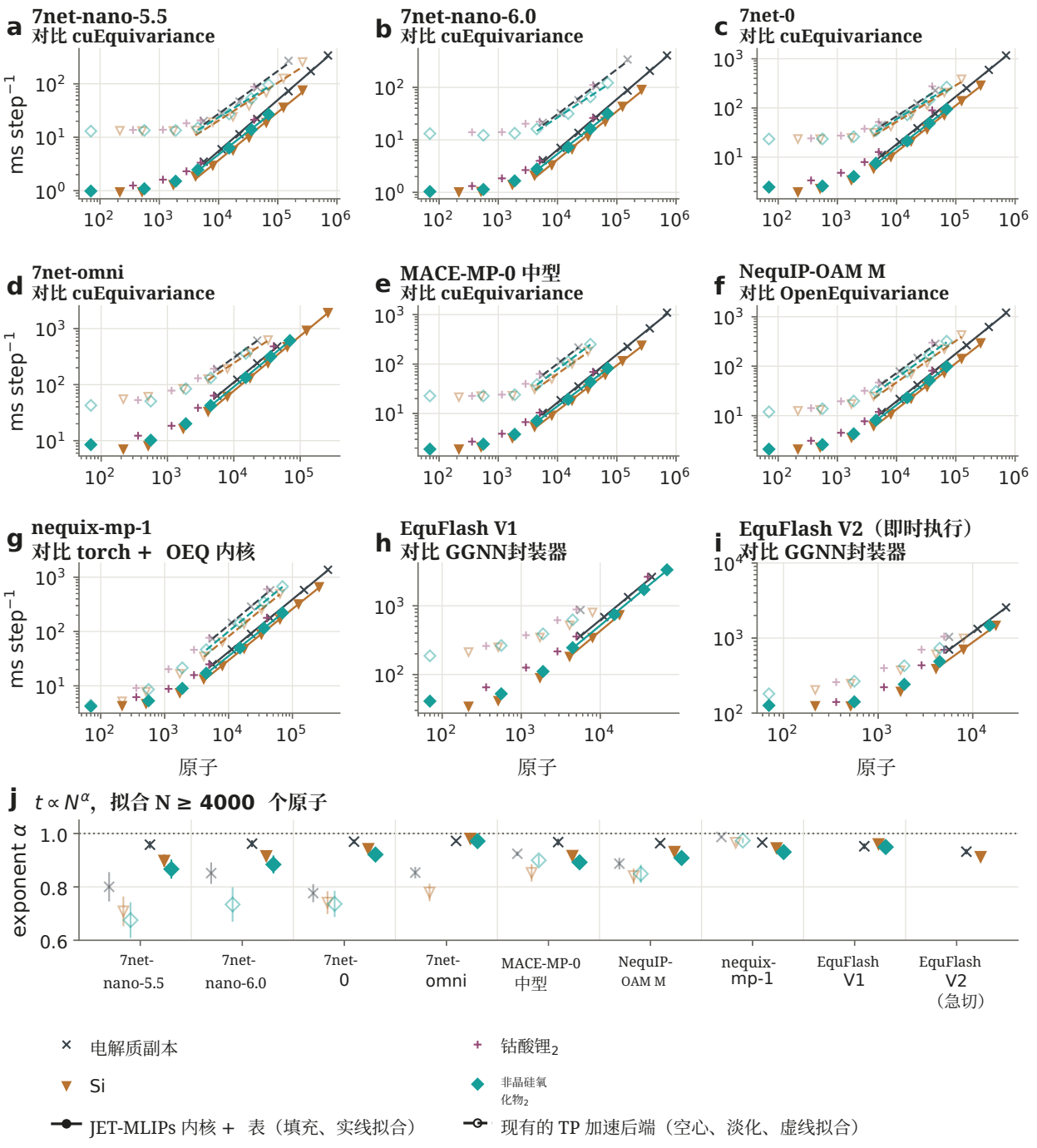
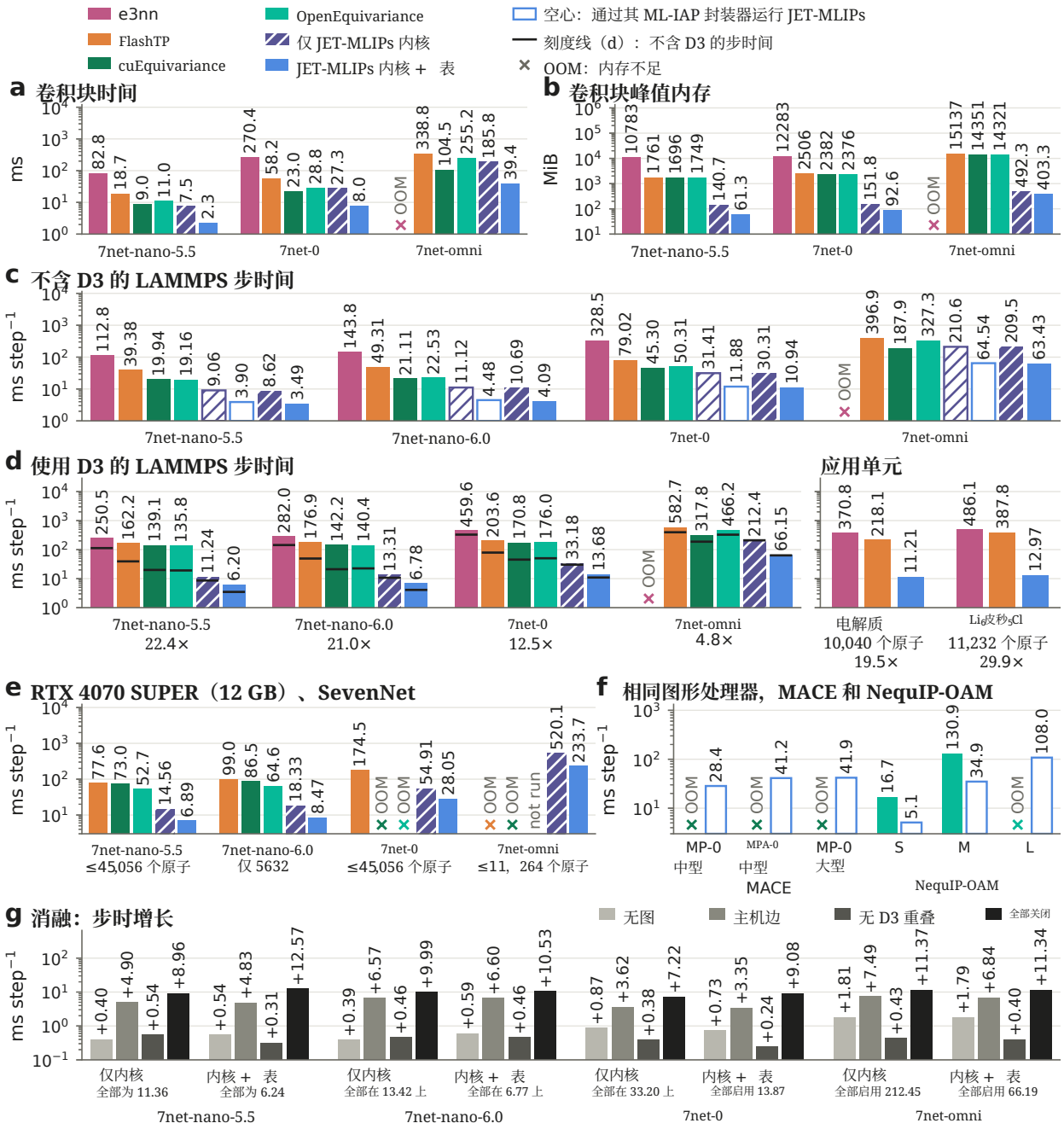


图 S8：步时间随体系规模的增长。(a–i) 对所标注模型，在使用 JET-MLIPs 内核 + 表和现有的 TP 加速后端时，LAMMPS 无需重建的步时间与原子数的关系（单个 H200、首次运行、电解质重复样本、$LiCoO_2$, Si 和非晶硅氧化物$_2$）。系统标记和后端填充/线型编码见图例。对 $N \geq 4000$ 个原子范围内的数据进行对数 $t = \alpha$对数 $N + c$最小二乘拟合，至少包含三个数据点，且不对 $LiCoO_2$ 进行拟合。(j) 这些拟合的指数 α及其标准误差；每个模型左侧的空心标记表示现有的 TP 加速后端，右侧的实心标记表示 JET-MLIPs 内核 + 表。点线标示 $\alpha = 1$。



图S9：SevenNet 后端、分子动力学集成和消费级 GPU 基准测试。 (a,b) 卷积块时间（前向加反向传播，各层求和）以及每层最大的峰值分配。(c) 不含 D3 的无重建 LAMMPS 步时间，其中实心 JET-MLIPs 柱使用e3gnn。(d) 含 D3 的无重建步时间（上游后端使用上游 D3，JET-MLIPs 使用 JET-D3），并标注 JET-MLIPs 内核 + table 相对于 cuEquivariance 的加速比；对于应用单元（7net-nano-6.0），还标注其相对于 FlashTP 的加速比。(e,f) 在不含 D3 的情况下，使用 12 GB RTX 4070 SUPER 并计入重建的平均步耗时，其中 (e) 标注了 JET-MLIPs 可容纳的最大系统规模。(g) 禁用单个e3gnn 集成功能或全部三个功能时，相对于全部启用配置的步时增长（以毫秒计）。(a–d) 和 (g) 使用一块 H200；除应用单元外，所有基准测试均使用 5632 原子电解质。

补充参考文献

[1]I.-B. Magdău、D. J. Arismendi-Arrieta、H. E. Smith、C. P. Grey、K. Hermansson 和 G. Csányi。分子液体的机器学习力场：碳酸乙烯酯/碳酸甲乙酯二元溶剂。《npj计算材料》，9:146，2023。doi: 10.1038/s41524-023-01100-w。[2]S. Grimme、J. Antony、S. Ehrlich 和 H. Krieg。94种元素 H–Pu 的一致且准确的从头算密度泛函色散校正（DFT-D）参数化。《化学物理学杂志》，132:154104，2010。doi: 10.1063/1.3382344。[3] S. Grimme、S. Ehrlich 和 L. Goerigk。色散校正密度泛函理论中阻尼函数的影响。《计算化学杂志》，32:1456–1465，2011。doi: 10.1002/jcc.21759。[4] A. Paszke 等。PyTorch：一种命令式风格、高性能的深度学习库。载于《神经信息处理系统进展》第32卷，2019。[5] NVIDIA。NVIDIA cuEquivariance。https://github.com/NVIDIA/cuEquivariance，2024。访问日期：2026-10。[6] V. Bharadwaj、A. Glover、A. Buluç 和 J. Demmel。面向 O(3)等变深度网络的高效稀疏核生成器。载于《2025年应用与计算离散算法会议（ACDA）论文集》，第32–46页。工业与应用数学学会，2025。[7] S. Y. Lee、H. Kim、Y. Park、D. Jeong、S. Han、Y. Park 和 J. W. Lee。FlashTP：用于机器学习原子间势的融合稀疏性感知张量积。载于《第42届国际机器学习会议论文集》，《机器学习研究论文集》第267卷，第33143–33156页，2025。[8] A. Johansson、E. Weinberg、C. Trott、M. McCarthy 和 S. Moore。LAMMPS-KOKKOS：跨百亿亿次级架构的性能可移植分子动力学。载于《SC '25研讨会论文集》，第1217–1232页，2025。doi: 10.1145/3731599.3767498。[9] C. R. Trott、D. Lebrun-Grandie、D. Arndt、J. Ciesko、V. Dang、N. Ellingwood 等。Kokkos 3：面向百亿亿次级时代的编程模型扩展。《IEEE并行与分布式系统汇刊》，33:805–817，2022。doi: 10.1109/TPDS.2021.3097283。[10] A. P. Thompson、H. M. Aktulga、R. Berger、D. S. Bolintineanu、W. M. Brown、P. S. Crozier、P. J. in 't Veld、A. Kohlmeyer、S. G. Moore、T. D. Nguyen、R. Shan、M. J. Stevens、J. Tranchida、C. Trott 和 S. J. Plimpton。LAMMPS——一种用于原子尺度、介观尺度和连续尺度基于粒子的材料建模的灵活模拟工具。《计算物理通讯》，271:108171，2022。doi: 10.1016/j.cpc.2021.108171。[11] J. Ekelund、S. Markidis 和 I. Peng。利用 GPU 提升迭代应用程序的性能：使用 CUDA 图进行内核批处理。载于《2025年第33届 Euromicro 并行、分布式与网络处理国际会议（PDP）》，第70–77页，2025。doi: 10.1109/PDP66500.2025.00019。[12] J. Kim、J. You、Y. Park、Y. Lim、Y. Kang、J. Kim、H. Jeon、S. Ju、D. Hong、S. Y. Lee、S. Choi、Y. Kim、J. W. Lee 和 S. Han。优化通用机器学习原子间势的跨领域迁移。《自然·通讯》，17:3432，2026。doi: 10.1038/s41467-026-70195-8。[13] J. Kim、J. Kim、J. Kim、J. Lee、Y. Park、Y. Kang 和 S. Han。面向高保真度机器学习原子间势的数据高效多保真度训练。《美国化学会志》，147(1):1042–1054，2025。doi: 10.1021/jacs.4c14455。

[14] D. Goldberg。每位计算机科学家都应了解的浮点运算知识。*ACM*计算调查，23:5–48，1991。doi: 10.1145/103162.103163。[15] S. Ehlert。Simple DFT-D3：D3色散校正的库优先实现。《开源软件杂志》，9(103):7169，2024。doi: 10.21105/joss.07169。s-dftd3 1.7.0版，https://github.com/dftd3/simple-dftd3。[16] S. Moganty、L. Abbate、A. Raghunathan、J. Sinicropi、G. Torres 和 Y. Wu。含磷电解质。美国专利10,665,899 B2，NOHMs Technologies Inc.，2020年。[17] S. Oh、J. You、J. Kim、J. Lee、H. An、S. Han 和 Y. Kang。通过知识蒸馏实现的轻量级通用机器学习原子间势，用于可扩展的原子模拟。*Journal of Chemical Information and Modeling*，2026。doi: 10.1021/acs.jcim.6c01103。[18] Y. Park、J. Kim、S. Hwang 和 S. Han。用于分子动力学模拟中图神经网络原子间势的可扩展并行算法。*Journal of Chemical Theory and Computation*，20(11):4857–4868，2024。doi: 10.1021/acs.jctc.4c00190。[19] L. Barroso-Luque、M. Shuaibi、X. Fu、B. M. Wood、M. Dzamba、M. Gao、A. Rizvi、C. L. Zitnick 和 Z. W. Ulissi。开放材料 2024（开放材料 2024（OMat24））无机材料数据集与模型，2024。arXiv:2410.12771。