arXiv:2611.30866v1 [physics.ao-ph] 23 Nov 2026

耦合天气与火灾模型的精确且可微重构

John S. Schreck[1,*]、John Dennis[1]、Iris Linck[1]、Maria Frediani[2]、Pedro A. Jiménez Muñoz[2]、Matthew Rehme[1]、Nihanth Cherukuru[1]和 David John Gagne II[1]

[1]计算与信息系统实验室，[2]研究应用实验室，美国国家科学基金会国家大气研究中心，博尔德，科罗拉多州，美国

*通讯作者：schreck@ucar.edu

摘要

人工智能正通过用学习模型取代物理模型，改变天气与气候预测。在此，我们展示了另一条路径：由人工智能编码代理在机器学习框架中重建物理模型本身，并逐位复现其预报结果。我们在 PyTorch 中重构了天气研究与预报模式的选定配置及其耦合火灾物理过程，并将与修正后 Fortran 代码的每一处差异都视为错误。在马歇尔山火模拟的 12 小时过程中，两套独立推进的代码在全部 365 个检查点上均逐位一致，涵盖大气、火灾和火炬粒子。当代码使用 Fortran 代码的数学库、一次执行一个操作时，这种一致性依然保持。在另一条快速路径上，重构后的模型在一块 GPU 上的运行速度约为 128核 CPU 节点上双精度 Fortran 代码的五倍。一次反向传播即可映射火灾对每个网格点风的敏感度。在合成测试中，其导数显示了应如何改变风和燃料湿度，以校正位置偏移的火线。在 24小时耦合预报中，神经网络取代辐射和火势蔓延计算，我们则针对物理计算测量其影响。这样一来，一个成熟模型在保留经过测试的计算过程的同时，变得快速、可微，并可引入学习组件。

引言

人工智能正在改变天气与气候预测[1–7]。在中期预报中，基于数据训练的模型如今已达到或超越领先物理预报系统的预报能力，并且可在图形处理器（GPU）上于数分钟内运行[1–5]。然而，这些模型是在利用物理模型生成的再分析资料上训练的，却搁置了蕴含数十年科学知识的方程和物理方案[8]。物理模型保留了这些知识，但其中大多数是为不适配此类硬件而编写的大型 Fortran 代码。将成熟模型移植到 GPU 上，将使这些计算能力用于基于数十年经验证物理知识构建的预报。

科学家如今希望从物理模型中获得的许多信息，很难通过 Fortran 代码获取。集合预报需要在决策可用的时间内进行大量模拟，而 GPU 可以让每次模拟更快[9,10]。变分资料同化通过观测估计鲜为人知的风或燃料湿度，需要计算预报相对于其输入的导数[11]。只有在耦合模型内部，才能评估用于替代物理方案的神经网络[12,13]。PyTorch 等机器学习框架[14] 提供 GPU 执行、自动微分，以及物理组件和学习组件的通用接口[15]。重写的检验标准在于，它能否提供这些能力，同时仍然复现既有的计算。

将既有模型迁移到这样的框架中，会重构软件。标量循环变为并行数组运算，状态也会重新组织[16]。这些变化可能改变数值计算，而看似合理的天气结果并不能有效检验翻译错误，因为错误的影响会随时间放大[17,18]。决定性的检验是精确复现既有计算[19,20]。例如，当我们用神经网络替代辐射方案时[21], ，可以将所得预报与同一模型中采用物理方案的预报进行比较。

天气研究与预报模式（WRF）提供了一项严苛的测试[22,23]。我们使用其耦合火灾组件 WRF-Fire[24,25], ，模拟 2021 年 12 月的马歇尔山火——这是迄今为止科罗拉多州损失最惨重的野火[26]。火灾热量会改变大气流动，进而影响火势蔓延以及火星（即余烬）的输送；这些火星可能引发新的火灾[27]。因此，在某个组件中引入的差异会在之后传递到其他组件，所以精确复刻必须重现每个组件及其间的每一次交互。

大语言模型（LLM）已用于翻译科学软件的单个部分[28], ，包括将其移植到可微框架[29,30]。大语言模型辅助 Fortran 翻译（LAFT）将翻译与数值验证和性能分析结合起来，用于一种微物理方案[31]。WRF 的独立导数代码（切线线性模式和伴随模式）支持变分资料同化，不过采用了简化的物理过程[32–34]。学习组件已与物理模型耦合[13,21]。据报道，移植到 JAX 框架的耦合模型在统计上保持一致[35,36]。据我们所知，尚无经过改写的天气模型能够逐位复现其 Fortran 代码的轨迹。

我们在此表明，人工智能编码代理能够在机器学习框架内重建成熟的物理模型，同时不改变其计算过程。我们开发并评估了 WRF-Speed——对部分耦合 WRF-Fire 配置的 PyTorch 重建版本（方法和补充表 S1）；该版本由人工智能编码代理编写，并通过编译后的 Fortran 参考实现进行核验。在逐项按原样执行操作的精确路径中，WRF-Speed 在马歇尔山火嵌套模拟的十二小时内，与修正后的 Fortran 参考实现逐位一致，且两段代码各自推进其预报状态。另一条快速路径采用标准 GPU 算术，不追求逐位一致；它在单块 GPU 上运行嵌套计算，速度约为 128核 CPU 节点上双精度 Fortran 代码的五倍。对耦合的马歇尔计算进行一次反向传播，即可映射火灾热释放对约 1100 万个网格值处风场的响应。经过核验的 WRF-Speed 导数能够在理想化合成测试中修正位置错误的火线。我们训练神经网络来复现物理方案，并在 24小时耦合预报中用这些网络替代辐射和火势蔓延计算，再将结果与物理预报进行比较。由此，成熟模型得以同时实现快速运行、可微分，并支持接入学习组件，同时保留经过测试的计算过程。

结果

组件测试中的逐位一致性

对单个模型组件（例如物理方案）的测试表明，WRF-Speed 使用其精确算术路径，能够逐位复现作为比较对象的 Fortran 输出。PyTorch 实现将 Fortran 中针对相互独立网格点的嵌套循环重组为可由 GPU 并行执行的数组运算。相互依赖的计算则保持原有顺序（图 1b）。

WRF-Speed 有两种运行方式。精确算术路径无需编译即可执行运算，并使用与 Fortran参考实现相同的数学库。另一种快速路径使用标准 GPU 算术，并将开销最大的运算整合到编译后的 GPU内核中，因此部分结果的末位可能存在差异。这两种路径都支持自动微分，可计算模型输出对指定输入的敏感度。我们会分别评估这些导数的准确性和适用性。

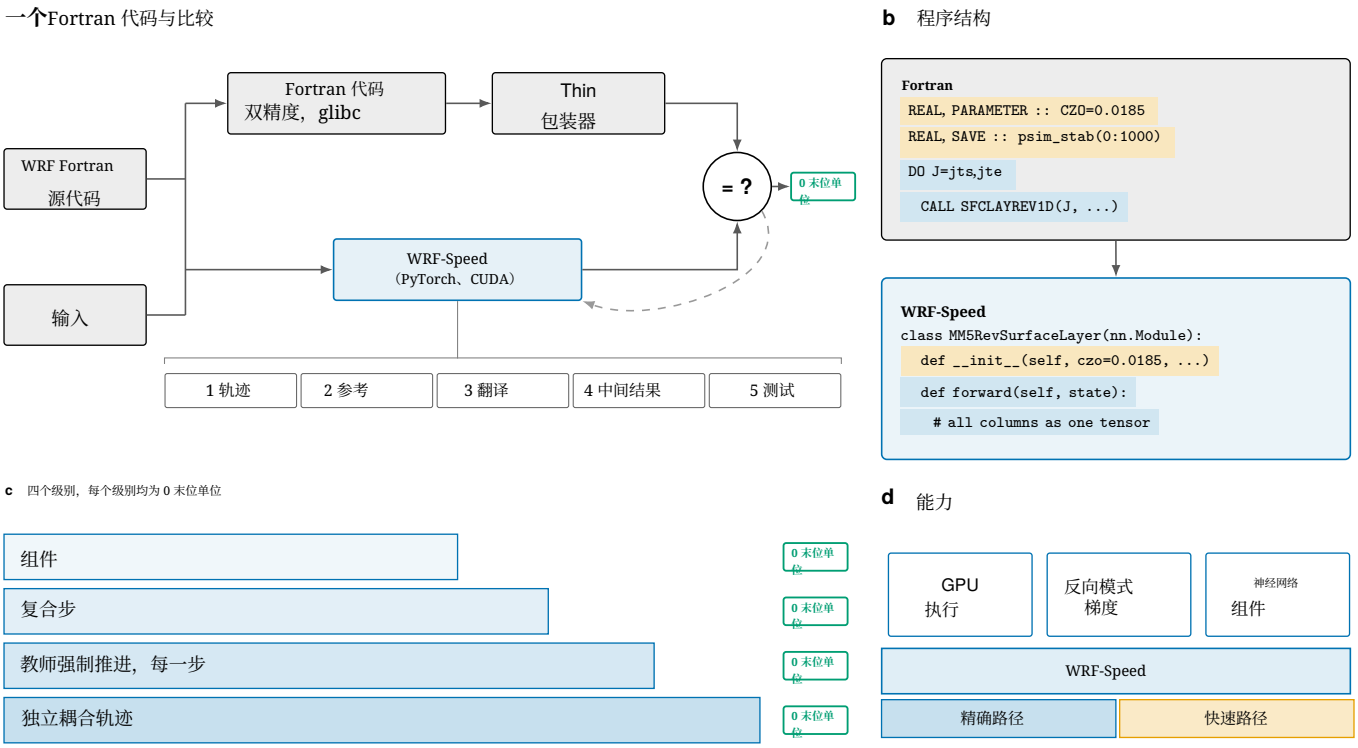


图 1：WRF-Speed 由 WRF Fortran 代码重写而成，并经过 0 ULP 检验。ULP 表示末位单位，即两个浮点数之间可能存在的最小差异。0 ULP 的一致性意味着两个值的每一位都完全相同。a，来自 Fortran 代码的转换。当结果不同时，我们会逐个操作查找首个出现不一致的操作（虚线循环）。带编号的方框标示每种方案所用的五个阶段。b，对应的近地表层代码，其中突出显示了匹配的操作。c，四个级别的比较。教师强制推进的每一步都从 Fortran 状态开始。d，能力和算术路径：用于与 Fortran 代码比较的精确路径（蓝色），以及用于计时、求导和神经网络预报的快速路径（橙色）。

组件比较涵盖近地表层方案的 29 个输出字段、Noah陆面模式的 42 个输出字段，以及 YSU 边界层方案的 12 个输出字段。比较还涵盖 WSM6微物理程序的 14 个输出、RRTMG长波辐射组件的 13 个通量和加热率输出，以及火灾计算在连续 36 个时间步中的水平集函数。使用 200 个粒子进行的测试涵盖火炬燃尽、终端速度、移除和沉积计算的全部八个输出。在这些测试中，Fortran 火炬代码的优化构建和未优化构建也会产生逐位完全相同的输出。

在六个理想化WRF测试案例中，WRF-Speed 在持续 3 至 12 个时间步的积分过程中，也重现了所有参与比较的 Fortran 字段。这些短时积分将比较范围从单个组件扩展到它们的联合执行。补充表 S4 列出了所报告的 Marshall 计算中使用的组件。

耦合 Marshall模拟中的逐位一致性

WRF-Speed 使用修正后的双精度 Fortran 代码和完全相同的算术路径，在模拟的 12 小时内重现了马歇尔山火模拟的大气场和火灾场（图 2）。我们比较了 1 千米父区域和 111米嵌套网格上的风、温度、气压和水分，并以 2 分钟为间隔比较火灾状态。在 365 个检查点中，

按位模式比较的全部 22,686,546,895 个浮点值均完全匹配。存储类型不同的数组在数值上也一致，因此每个检查点比较的数组值总数约为 6,800 万。每次比较中，活跃火星的物理属性都一致。两个代码各自推进大气和火灾状态，因此，这项比较验证了模拟演进过程中的耦合与反馈。

这一致性表明，重写后的程序保留了经测试的耦合计算，包括组件测试可能遗漏的交互作用。与观测到的最终周界相比，两种代码都高估了火灾面积。单独的快速路径运行采用标准 GPU 算术，并遵循不同的轨迹（图 2g–i、k）。下面，我们通过在单个实现中扰动风，考察对微小数值变化的敏感度。

飞火方案从燃烧单元格释放粒子，使其随三维风输运，计算燃烧和沉积，并在满足着陆条件时点燃新火灾。飞火会加速火灾蔓延，使其能够越过防火带，并且在风驱火灾中发挥主导作用。Fortran 实现会在 MPI 进程之间交换粒子数组（即在 CPU 集群上共享工作量的独立进程），并且必须在重启过程中保留粒子状态。物理过程及其数值模型表示具有高度非线性。粒子释放高度或下落速度的单个位差异，就可能改变新点火的位置，导致两次运行的状态出现显著分歧。因此，重现这些事件需要一致的粒子计算和状态传递。

火蔓延体数量达到 87,093 个活跃粒子，并且在每次比较中，它们对应的物理属性都与 Fortran 代码一致（补充图 S2b）。对 Fortran 代码的校正中，超过一半涉及火蔓延体计算（补充表 S6）。例如，在没有粒子的进程排名上，一个未初始化变量导致 Fortran 结果在不同运行之间出现差异。

对耦合模型的比较发现了各组件测试未能发现的值传递误差。在一个案例中，嵌套区域重新计算了自身的臭氧廓线，而没有使用从父区域插值得到的廓线。向下长波（红外）辐射的相对变化最多为 2.7×10^{-5} in，但经过 34 个父步后，超过一半的大气值（2.46 亿个中的 1.36 亿个）出现差异。在另一个案例中，辐射组件将加热率传递给动力核心时遗漏了柱质量因子，导致辐射加热约小了 10^5 倍，尽管辐射例程与 Fortran 代码一致。因此，必须将耦合模型作为一个整体进行测试，因为连接代码中的错误可能改变整个预报，即使每个组件仍然都与 Fortran 代码一致。

GPU 性能与导数检查

将计算写成数组运算，可让 GPU 并行处理彼此独立的网格单元和大气列。快速预报还必须避免临时数组，以及启动大量小型 GPU 运算所带来的开销。因此，我们将选定的计算合并为更大的 GPU 内核（在 GPU 上运行的小型程序），使用 CUDA 和 Triton[37] 编写，或由 PyTorch 编译器[38]生成。我们将每个时间步的内核执行序列记录一次，并将其作为一个整体重放，称为 CUDA 图。重放会在当前模型状态上重复相同的运算序列，因此每次重放的时间步仍会计算出新的状态。执行辐射更新的时间步不使用重放。

在嵌套预报的 100 个父步中（包括五次辐射更新），快速路径在一块 A100 80 GB GPU 上以双精度运行，平均每步耗时 1.166 秒；该值为两次运行的平均值（图3c）。不进行辐射更新的时间步平均耗时 1.118 秒，进行辐射更新的时间步平均耗时 2.084 秒。在一个 128核 CPU 节点上，Fortran 双精度运行的平均耗时为 6.075 秒，生产环境所用单精度运行的平均耗时为 2.769 秒。在这段模拟的五分钟内，GPU 计算吞吐量分别是双精度 Fortran 和生产环境单精度 Fortran 的 5.2 倍和 2.4 倍。这两种

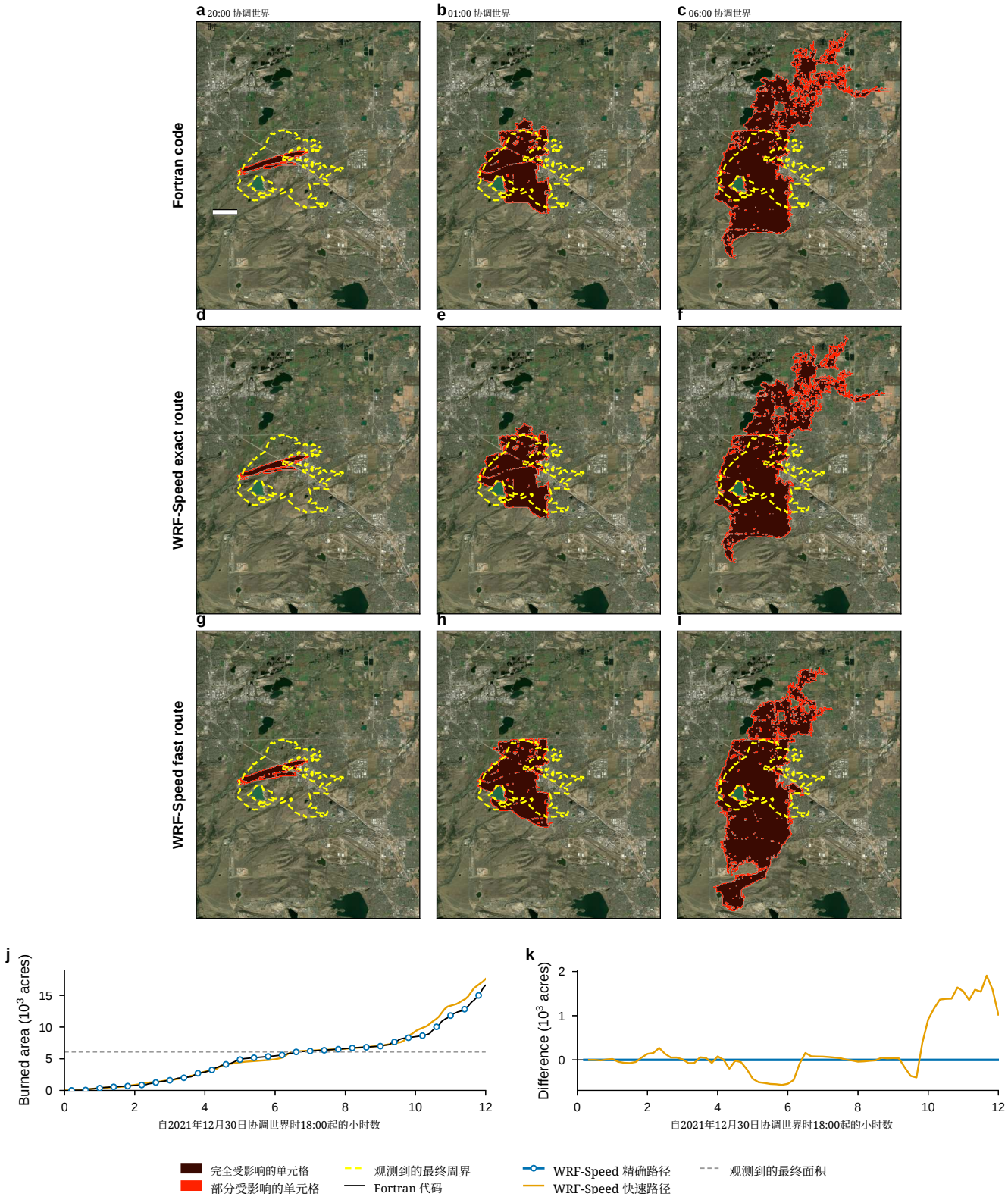


图 2：WRF-Speed 精确路径逐位复现了 Fortran 代码中的马歇尔山火轨迹。a–i，Fortran 代码（a–c）、WRF-Speed 精确路径（d–f）和单独快速路径运行（g–i）中的完全受影响（深红色，燃烧比例至少为 0.999）和部分受影响（亮红色）火灾网格单元。各列分别显示 2021年12月30日协调世界时 20:00，以及 12 月 31 日协调世界时 01:00 和 06:00。黄色短线标示观测到的最终火场边界。比例尺：2 千米。j，Fortran 代码（黑色）、精确路径（蓝色圆圈）和快速路径运行（橙色）的过火面积。虚线表示观测到的最终面积。k，每次 WRF-Speed 运行的过火面积减去 Fortran 代码的过火面积。精确路径的 5 个差值（蓝色）在每个检查点均为零。底图：Esri World Imagery。

代码从单独生成的重启状态开始，计时不包括启动和输出。最终的 WRF-Speed 状态与先前一次未进行重放的运行所保存的状态逐字节一致。

我们还在 111米嵌套网格上对九个火耦合子步进行导数计算，同时固定来自父区域的强迫和辐射加热率。我们测量这三秒结束时的平均湿位温如何随初始风场和温度的均匀偏移量而变化。前向传播推进模型，反向传播计算导数。两者在 A100 上总计耗时 13.73 秒；在此过程中，我们将选定的中间状态存储在 CPU 内存中（图 3a、b）。使用不同的软件构建版本并将中间状态保留在 GPU 上时，GB200 的运行耗时为 5.76 秒。在这两种配置下，状态、输出和导数均与相应的未编译 PyTorch 计算逐位一致。这种一致性验证的是快速实现，而不是导数的准确性。

我们通过一项单独的实验检验导数的准确性：略微改变一个输入，重新运行模型，并将输出的变化与导数进行比较（有限差分检验）。在 Marshall嵌套区域的一个子步中，过火面积关于风的导数在预先选定的五个风增量下均未达到容差要求。在后续运行中，当增量缩小到 0.00001 至 0.005 $\mathrm{m\ s^{-1}}$ 范围内的五个数值时，它在每个增量下都达到了容差要求。当我们固定模型的条件选择——也就是代码在备选方案之间进行选择的位置，例如风从网格单元的哪一侧到达——时，在原先的五个增量中，它还在其中四个增量下达到了容差要求（方法）。

火灾对三维风的敏感性

接下来，我们计算耦合的 Marshall 模拟对每个网格点风的微分。我们从2021年12月30日协调世界时19:15开始运行111米嵌套网格，此时点火刚刚越过美国36号公路；随后推进九个子步（模拟3秒），并启用大气、火灾、火星以及火灾–大气反馈。输出为3秒结束时火灾的总热释放（0.89 GW）和过火面积（1.60 $\mathrm{km^2}$）。某个网格点的敏感度表示该点初始东西向风或南北向风每变化1 $\mathrm{m\ s^{-1}}$ 时，输出的变化量。若通过重新运行模型来估算这些数值，则每个风分量都需要针对约540万个数值各进行一次模拟。相比之下，反向模式微分（即机器学习中的反向传播，或资料同化中的伴随法）只需针对每个输出进行一次反向传播，即可得到所有这些数值。

图4显示，点火处的敏感度在接近地面的位置最大。对于两种输出和两个风分量，最大值位于相邻网格点的模型第三层（图4a、c、d）。南北向风在此处占主导地位。南北向风的最大热释放敏感度为 2.06×10^{7} W per $\mathrm{m\ s^{-1}}$（单个网格点处每 $\mathrm{m\ s^{-1}}$ 总热释放的2.3%），而东西向风为 4.18×10^{6} W per $\mathrm{m\ s^{-1}}$。相应的过火面积值为27.0和4.37 $\mathrm{m^2}$ per $\mathrm{m\ s^{-1}}$。在点火周围，白色箭头大多指向北方（图4c、d）。具体而言，点火是一条与西风气流一致的狭长带状区域（图4b），而向北分量会使风横穿其北侧长边，吹向未燃燃料。

对嵌套网格内的敏感度求和，可得到风均匀变化时的响应。东风每增加 $\mathrm{m\ s^{-1}}$，热释放就增加 31 MW，从而增强现有的西风气流。南北向总和为负值（每 $\mathrm{m\ s^{-1}}$为 -42MW），因为敏感度在嵌套网格内变号，且符号相反的区域影响超过了点火。燃烧面积总和的符号相同（$+32.9$ 和每 $\mathrm{m\ s^{-1}}$为 $-19.4\mathrm{m^2}$ ）。

请注意，约有 38% 的风速值会在 3 秒内影响输出，其影响远超过火面积。风速变化会通过声波传递的压力调整，以及输运计算在每个龙格–库塔阶段中传播至多个相邻网格单元，从而在模型中传递。这些远距离影响的数值很小，而包围幅值最大 3% 数值的壳层始终位于火灾上方，并随其羽流上升（图 4a）。我们尚未将这些

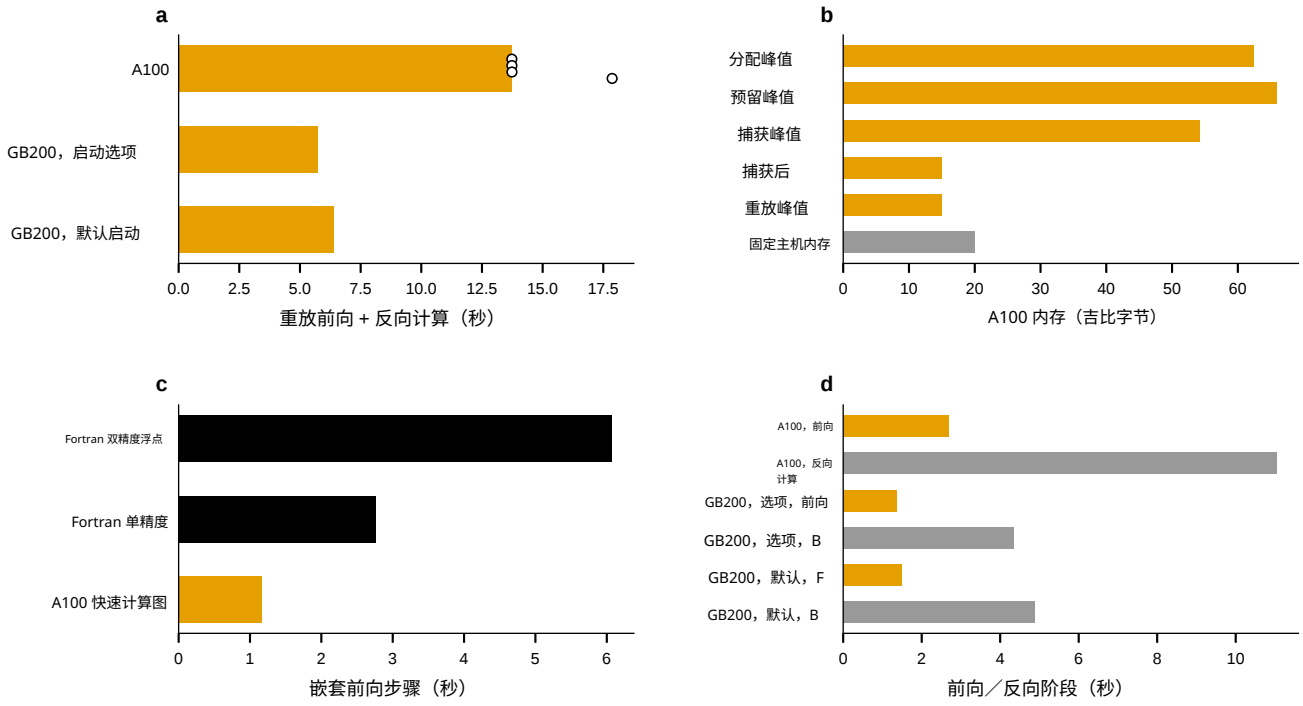


图 3：在一块 GPU 上，快速路径执行一个嵌套预报步的速度约为双精度 Fortran 代码的五倍。a，在父级强迫和辐射趋势项保持不变的情况下，对 111米嵌套网格的九个子步（模拟 3 秒）进行一次前向和反向传播的中位墙钟时间。中位数涵盖四次 A100 或八次 GB200 重复运行，空心圆表示四次 A100 重复运行结果。两个 GB200 柱分别表示使用和不使用一组选项运行的结果。b，该传播过程中的 A100 内存。橙色柱表示 GPU 内存，灰色柱表示存储在 CPU 内存中的中间数组。c，在 128核 CPU 节点（黑色）上，双精度和单精度 Fortran 构建版本执行每个嵌套父步的平均墙钟时间（仅前向计算和包含辐射更新），以及在一块 A100（橙色）上使用 WRF-Speed 的相应时间。d，a 中传播过程的前向（F，橙色）和反向（B，灰色）部分。GB200 的前向和反向时间来自与 a 中墙钟时间不同的另一次运行。所有 GPU 计时均使用快速路径。

单个数值与有限差分结果进行比较。在这个嵌套网格上，过火面积导数仅在风速变化不超过 0.005 ms^{-1} 时，或在固定代码中的条件选择时，才与有限差分结果一致（方法）。因此，每个数值表示的是 3 秒内的局部变化率，而不是对 1 $\mathrm{m\ s}^{-1}$ 变化的预测。即便如此，只需一次反向传播，就能确定哪些风速观测或校正最有可能改变火灾预报。

火线敏感性对燃料湿度的响应

导数可预测燃料变得更干燥时火线移动的距离。通过一次预报，它可以给出燃料湿度发生微小变化时锋面位置的变化，随后可由新模拟进行检验。我们采用了比 Marshall 案例更简单的设置，因此无需考虑地形、火星和完整物理过程，就能清楚地观察单一输入的影响。计算域为平地，宽 2 千米，模型以 0.25 秒为步长推进。火灾与大气仍保持耦合，只有初始燃料含水率发生变化，而干燃料载量保持不变。我们沿五条固定线追踪火头，即其前沿（方法）。

起始于　　在水分为 8% 时，预报中的导数表明，火头在 7% 时移动得更快　　%

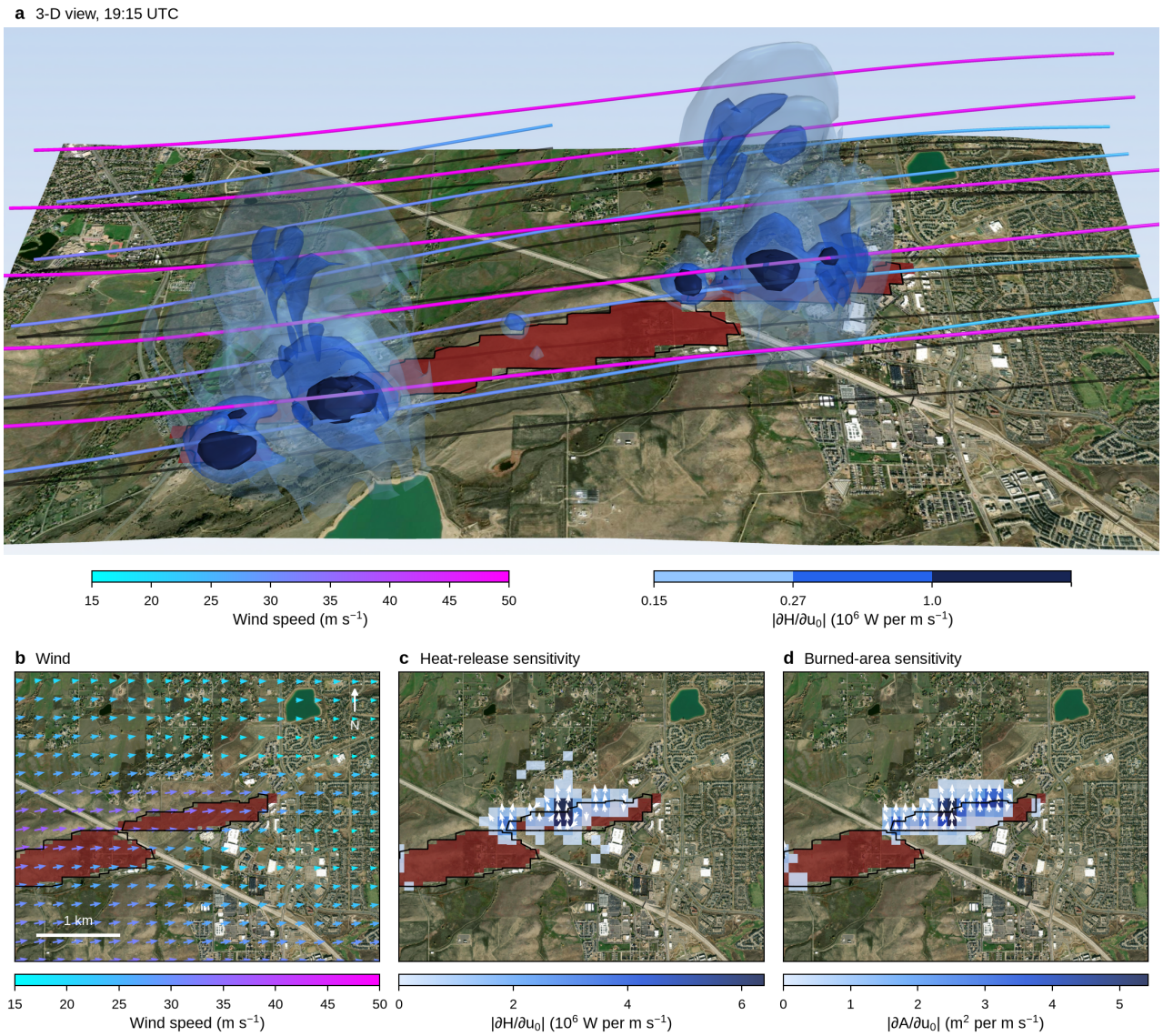


图4：马歇尔山火的热释放和过火面积会对三维网格中的风作出响应。场数据来自协调世界时2021年12月30日19:15的111米嵌套网格。每个敏感度表示在一个网格点处，初始水平风每变化1 m s^{-1}，3个模拟秒后总热释放（0.89 GW）或过火面积（1.60 km^2）的变化。a，北向朝上、垂直夸大2.5倍的三维视图。蓝色壳层圈出热释放敏感度绝对值最大的3%、0.7%和0.1%。流线沿风向延伸，并按水平风速着色（15–50 m s^{-1}）。过火面积以深红色显示，边缘为黑色。b，模式第三层上的风。箭头指向下风向，并按风速着色，色标与a相同。c，模式第三层上的热释放敏感度绝对值（W per m s^{-1}）。白色箭头指示在该点增加风速最能提高热释放的方向。d，与c相同，但显示过火面积（m^2 per m s^{-1}）。b–d面板放大了越过美国36号公路的点火区域。底图：Esri World Imagery。

而在 9% 时移动得更慢。60 秒（240 步）后，这两种变化使火头分别向相反方向移动约 1 米，而该预报与新模拟的结果相差不超过 0.09 米（图 5a–c）。中心有限差分通过对水分略高和略低时的模拟结果进行估算来求得导数，

在三个预报时刻分别验证导数本身。因此，一次求导的预报即可同时给出干燥和湿润响应。

导数给出的直线预测仅在起始点附近成立。当水分含量为 6% 和 10% 时，其误差增至火头位置模拟变化量的 16–18%（补充表 S2），这表明水分响应呈曲线变化。因此，这项实验既测量了对水分的敏感性，也针对该案例考察了单个导数在偏离 8% 多远时仍然有效。

基于梯度的合成火线校正

导数还可以指出，如何调整火线位置有误的预报。根据观测到的火线估算风场或燃料湿度，是野火建模中的成熟做法[39–42]。在此，我们计算每个锋面位置对两个初始风偏移量及燃料湿度变化的响应。这些导数组成一个矩阵，即火线雅可比矩阵；每一行对应一个锋面位置，每一列对应一个输入。我们从两组不同的风偏移量和水分初始猜测值出发，并将其称为起始点。我们用不同版本的 Fortran WRF 生成的含噪合成火线拟合这三个输入；该版本的时间步长是我们所用时间步长的一半，火线对应的时刻为 120 秒和 180 秒。在进行更新之前，我们将锋面位置及失配的导数与有限差分结果进行核对。

一次高斯–牛顿更新利用雅可比矩阵减小失配。在 300 s 时——这个时间点未用于拟合——火线位置的均方根误差，从第一个起始点的约 52 米降至 9 米，从第二个起始点的 28 米降至 7 米（图 5d–i,l）。沿着固定射线，我们还可以看到校正后的火线仍位于目标之内还是之外。两个起始点的火头处大部分偏差都消失了，剩余误差则随方向而异（图 5j,k）。

从两个起始点出发，仅改变风比仅改变水分更能减小后续火线误差（图 5l 和补充表 S3）。从第二个起始点出发，仅调整风的更新使后续误差达到最小，约为 3 米。同时调整风和水分时，两个起始点会得到不同的组合，但对火线的拟合效果相近（图 5m–o）。这两种组合都无法还原生成这些火线的参数值。因此，校正改善了火线位置，却无法识别其背后的风和水分。

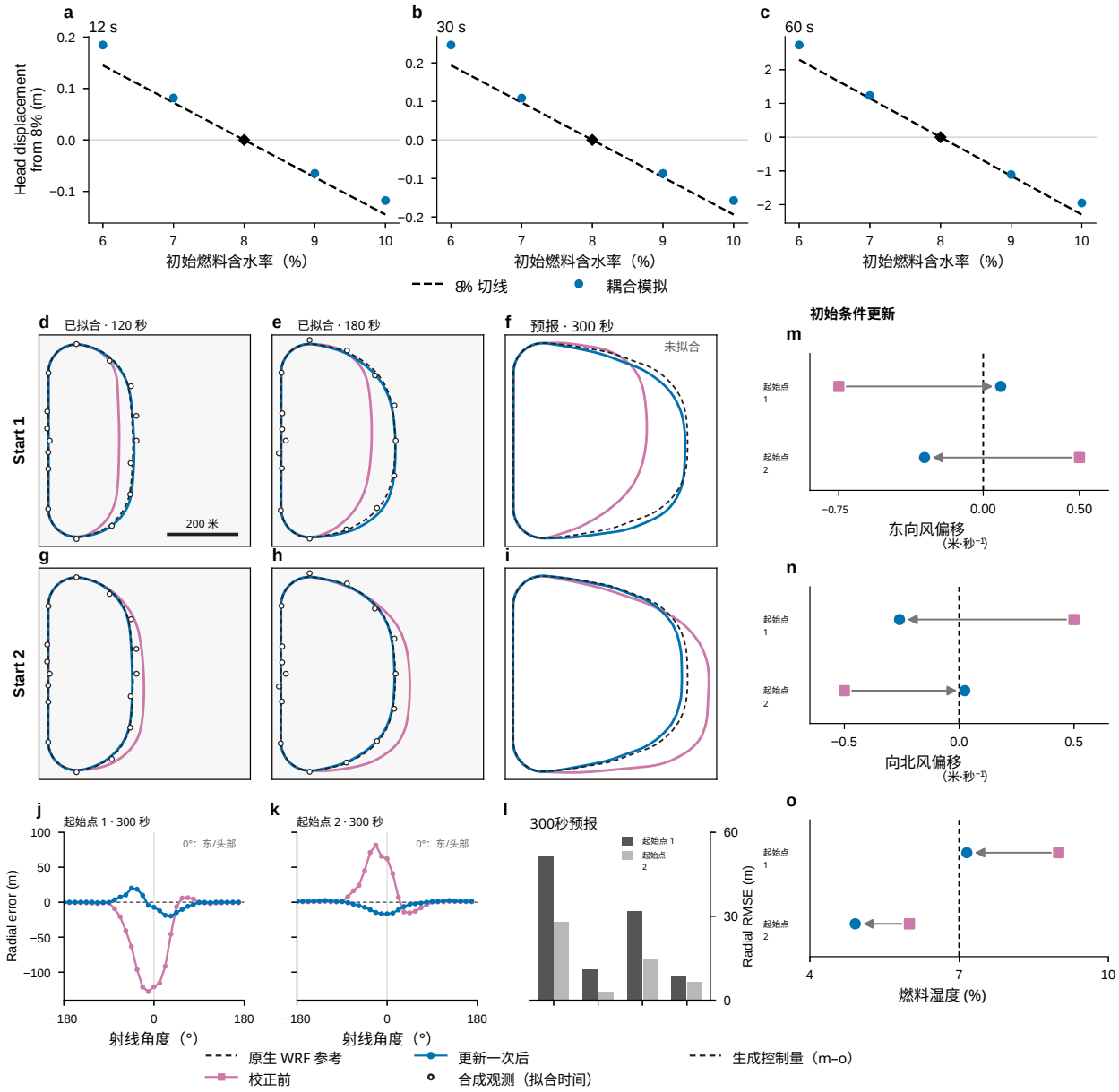


图 5：导数可预测燃料湿度对火灾响应的影响，并在合成测试中校正偏移的火线。a–c，12、30 和 60 秒时，火头相对于 8% 湿度预报的位移。虚线表示根据 8% 湿度下的导数作出的预测。蓝色圆圈表示新进行的耦合模拟。d–i，第一组（d–f）和第二组（g–i）起始点在 120、180 和 300 秒时的火线：来自原始 WRF 的目标火线（黑色虚线）、校正前的火线（粉色）以及进行一次高斯–牛顿更新后的火线（蓝色）。圆圈表示已拟合时刻的合成观测。300秒预报未拟合。比例尺为 200 米。j、k，32 条固定射线上的 300 秒有符号径向位置误差：预报半径减去目标半径，其中 0° 向东、 90° 向北。l，两组起始点的 300秒预报径向均方根误差（RMSE），比较校正前以及仅更新风、仅更新水分或联合更新后的结果。m–o，箭头连接初始输入（粉色方块）与更新后的输入（蓝色圆圈）；这两类输入分别为两个风偏移量和燃料湿度。虚线标示生成值。

采用神经网络组件替代的耦合预报

神经网络可以替代耦合预报中的物理方案，而预报结果能显示每项替代带来的变化。每个网络输出的量与被替代的物理方案传递给模型其余部分的量相同。我们选择辐射，是因为其计算成本较高；选择火势蔓延，则是因为它在物理上十分重要且有改进空间。作为示范，也为了探索基于观测进行训练的起点，我们使用被替代物理方案的输出训练网络。五种配置分别替代短波（太阳）辐射、火势蔓延、短波辐射和火势蔓延两者、长波辐射，或三者；每种配置都进行24小时预报（图6）。物理预报和神经网络预报从相同的输入和随机数状态开始，随后独立推进，同时启用大气反馈和火星机制。

我们翻转初始风场的最后一位（一个末位单位），以展示在所有物理方案保持不变时，火势增长可能出现多大差异。这些翻转没有物理意义，但随着风、热释放和单个火星事件相互作用，它们会演变出不同的火灾演化历史。因此，我们将每个神经网络预报与一组扰动物理预报以及未扰动的物理预报进行比较，从而使过火面积和火场位置的差异处于模型数值敏感度的尺度上。

短波网络逐层向下传递阳光。长波网络预测向上和向下通量，这些通量决定每层的升温或降温幅度。我们还使用 Marshall 模拟中选定的列训练了长波网络，因此这次预报重用了部分训练案例。蔓延网络学习取决于燃料属性、风和坡度的速率因子。燃料和含水量预处理以及蔓延方向仍属于物理计算的一部分。该网络还采用了不同的火锋计算实现，因此比较结果同时包含这两项变更的影响。方法和补充材料S4节介绍了训练细节。

图6追踪了从第一个小时开始的烧毁区域，并展示其面积变化历程，同时列出扰动物理预报。我们在相同的预报时刻，并在共享初始状态的网络与物理预报配对中，通过过火面积、火灾网格上的足迹重叠以及大气场，分别评估每项替换单独使用和组合使用时的效果。完整的变化历程至关重要，因为随着火灾演变，面积差异的正负和大小都会变化，而且相似的总面积可能对应不同的火线位置。

仅改变初始风场的最后一位，十个物理预报的24小时过火面积便达到187.5–371.5 km^2 。我们以相同方式将每种网络配置运行成包含十个成员的集合，五种配置的结果均落在这一范围内（图6d）。与物理预报的平均过火面积相比，短波配置高出2.9%，蔓延配置高出4.0%，短波与蔓延联合配置高出16.2%，长波配置高出26.1%，三者配置高出9.3%。配对的历史记录展示了过火面积差异随时间的变化过程（补充图S3）。

相近的过火面积可能对应不同的影响范围。绘制出的长波预报过火面积仅比绘制出的物理预报多1.6%，但两者对任一预报中过火区域的重合率仅为71%。作为尺度参照，我们另外运行了四个扰动物理预报，这些预报与十成员集合相互独立；它们与同一物理预报的重合率为65–78%。

大气的差异比过火面积更大。在仅替换短波、长波以及三者全部的预报中，风、地表温度以及地表热量和辐射通量的差异，均超过了这四种物理比较所得的相应范围（补充表 S9）。补充章节S5报告了另一项实验，其中由精度较低的网络替代了部分边界层方案。

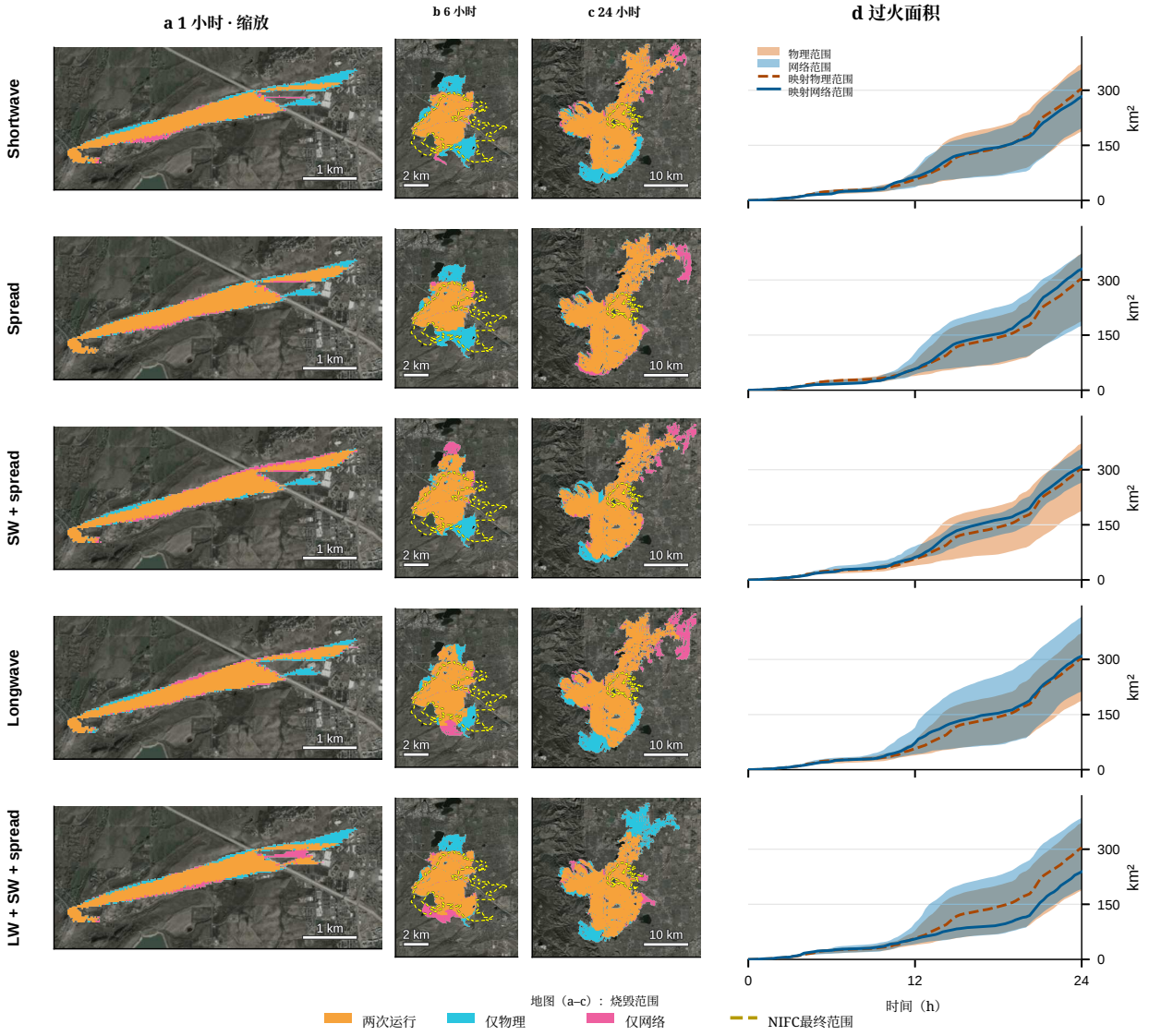


图6：神经网络组件在耦合火灾预测中分别运行并协同工作。各行分别展示使用短波、蔓延、短波和蔓延、长波以及三者网络的预报。a，一小时火场范围，放大显示。b，六小时火场范围。c，二十四小时火场范围。地图颜色表示过火重叠区域：橙色表示两次运行均覆盖；青色表示仅物理；粉色表示仅网络。黄色短划线标出国家跨部门消防中心（NIFC）映射的最终范围，作为地理背景。d，过火面积变化历史。橙色（物理）和蓝色（网络）阴影分别表示各十个24小时预报的范围。物理成员之间的差异仅在于初始风场的末位比特变化。橙色虚线和蓝色实线曲线分别标出a–c中映射的物理预报和神经网络预报。每一行中映射的物理运行均为同一个物理预报。底图：Esri World Imagery。

讨论

成熟的物理模型可以在不改变计算的情况下移植到机器学习框架中。由于计算十分敏感，数值基准必须精确。火星若在下风向多落一个网格单元，就可能在不同地点引发火灾。在 Marshall 模拟中，修正后的 Fortran 代码与 WRF-Speed 在十二小时内的每个检查点上，对风、火势增长和火星数量的模拟结果都一致。两者的火势都蔓延到了观测到的最终周界之外，因此可以排除翻译过程是它们共同出现过度蔓延的原因。我们还将成熟的计算用作数值基准[18]，以检查编码智能体编写的代码。即使周边代码传递给大气的加热量不足，辐射例程本身仍通过了测试。将组装后的模型与 Fortran 代码进行比较，揭示了这一误差。

两种不同的风与燃料湿度组合，都能较好地拟合同一条火线。在理想化实验中，只需对风和燃料湿度进行一次调整，就能改善对较晚时刻火线的拟合，而该时刻的数据未用于拟合。然而，这两个起始点会导出不同的风和水分状态，且两种修正后的预报都能很好地拟合较晚时刻的火线。因此，火场边界有助于修正预报，却无法确定形成该火线的风和水分状态。这一区别对于根据观测到的火线估算风和水分的工作至关重要[40,42]。

网络已与大气求解器耦合，并共同用于湿物理过程和辐射[12,13]。在耦合火灾预测中，我们还可以追踪单项替换和组合替换的影响，直至火场边界。长波网络展示了这样做的价值。它的预报与扰动物理模型所得总过火面积的范围有所重叠，但其集合平均值高出四分之一。单次预报也可能在不同位置产生相近的过火面积。这些扰动为每项比较提供了一个尺度，该尺度由物理模型本身的敏感度决定。

精确比较涵盖一个嵌套火灾案例。计时和神经网络预报均采用快速路径。理想化导数通过了检验，而 Marshall 的结果取决于所测试的风扰动。根据观测到的火场边界进行更新时，还需要考虑观测和模型两者的误差。我们训练这些网络来复现现有方案。它们对观测到的火灾的预报准确度有何影响，仍有待检验。

Rothermel 火势蔓延公式可追溯至 1972 年，用于描述准稳态地表火蔓延[43]。在观测到的火灾上训练的网络可以取代它，并像这里的蔓延网络一样，在完整预报中接受检验。其他正在重写成熟模型的领域，也可以用同样的方式利用现有计算，既识别其中已有的误差，也衡量新方案取而代之时带来的变化。因此，成熟模型经过检验的物理机制与机器学习工具可以兼得。

方法

我们首先介绍重写的模型配置（代码库与范围）。为了让编码智能体的工作可供核查，我们添加了简短程序，以最后一位数字的精度打印每个 Fortran例程的输入、中间值和输出，因此每项检查都是针对编译后的 WRF 代码本身，而不是手写的仿制版本（参考构建）。为了进行精确比较，我们使用双精度，并在两份代码中使用相同的数学库（glibc）计算指数函数和对数函数等[44]（WRF-Speed算术）。我们分五个经过核查的阶段转换每个物理方案（转换流程）。我们将与 Fortran 代码的任何差异，无论多小，都视为错误；通过为每个例程提供相同的输入，并找出第一个结果不一致的例程，来确定错误来源（数值差异诊断）。随后，我们在两份代码中运行完整的马歇尔山火配置，并每隔两个模拟分钟比较一次保存的状态（耦合比较）。确认两者一致后，我们在 GPU 上对预报及其导数进行计时

（计时）。我们通过有限差分检验导数（梯度检查），并用它们校正位置偏移的火线（火线校正实验），以及预测火线对燃料湿度的响应（燃料含水率响应实验）。我们还介绍过火面积及其他绘图量的计算方法（诊断计算与图件制作）。最后，我们训练神经网络来替代物理方案，并在耦合预报中运行这些网络（神经网络组件训练与耦合评估）。

代码库与范围

WRF 的动力、物理和框架源代码约有 80 万行 Fortran。我们使用 PyTorch 重写了部分 WRF 配置，并将重写结果与以双精度编译的 Fortran 代码进行了核对。该 Fortran 代码包含已记录的校正（参考构建）。补充表 S4 列出了重构的组件、各组件的计算内容及其 Fortran 源代码规模，总计约 13.1 万行。补充方法的 S1 节介绍了实现和执行配置。

主要比较采用马歇尔山火模拟的配置[27]。其动力核心为 WRF-ARW，每个时间步分为三个龙格–库塔阶段推进，并在每个阶段内以更短的步长处理声波（声学子步）[22,45]。垂直坐标在近地面处随地形变化，并随高度逐渐变平（混合质量坐标[46]）。1 千米外层域（父区域， $163 \times 159 \times 44$ 个网格单元，3秒步长）为 111 米内层域（嵌套网格， $360 \times 342 \times 44$ 个单元）提供边界值。每个父区域步长内，嵌套网格执行九个 1/3 秒步长，称为嵌套子步。物理方案包括 YSU 边界层方案[47] （仅用于父区域）、修订后的 MM5 地表层[48], 、Noah 陆面方案[49], 、WSM6 微物理方案[50], 、RRTMG长波辐射[51] 和 Dudhia短波辐射[52] 。火灾模型为 WRF-Fire[24,25]。它使用水平集函数追踪火线；该场在火灾已燃区域为负值，在火线上为零。火线以 Rothermel 蔓延速率[43], 向前推进；该公式基于燃料属性、风和坡度，并采用 Anderson 燃料类别[53]。火灾网格在两个水平方向上的分辨率均为 111 米嵌套网格的四倍。火星飞散方案[27] 将单个火星（由风携带的燃烧余烬）作为粒子进行追踪。

参考构建

Fortran 代码是采用 GNU Fortran 以双精度编译的 WRF 4.5.1。我们让编译器调用指数函数和对数函数等函数的标准逐值处理版本（标量函数），因为一次处理多个值的更快版本可能会产生不同的舍入结果。我们在编译火灾飞散物代码时禁用自动循环向量化，这是一种能够同时处理多个循环迭代的编译器优化，但可能改变结果的最后一位。为保存中间值，我们在 Fortran 代码的副本中添加输出语句。在使用每个修改后副本的输出进行比较之前，该副本必须逐位复现未修改代码的计算结果。用于比较的 Fortran 代码包含有文档记录的校正（补充表 S6）。其中半数以上涉及火灾飞散物代码，其余涉及陆面方案、火灾风场插值和重启。耦合的 Fortran 代码在 4×8布局中运行于 32 个消息传递接口进程上。

WRF-Speed算术

精确复现要求执行完全相同的舍入操作序列[44]。因此，我们保留 Fortran 代码中每个表达式的原有分组方式。Fortran 代码在哪里执行除法，我们就在哪里执行除法；在哪里乘以倒数，我们就在哪里乘以倒数，因为两种运算的舍入结果不同。我们还按照相同顺序累加每个求和式中的各项。彼此独立的网格点和列，以及集合中的各个独立预报，都可以并行运行；而相互依赖的计算则保持其

顺序。这些计算包括垂直扫描（每一层都使用前一层的结果）、龙格–库塔阶段和声学子步。我们按照 WRF 读取各项配置值时所用的精度进行复现。例如，WRF 从 namelist（其运行配置文件）中以双精度读取阻尼系数。相比之下，它以单精度存储 Noah 土壤层厚度，然后再将其转换为双精度。

我们使用两种算术路径。在精确路径中，PyTorch 会在每个操作被调用时立即执行，不进行编译。GPU 版本的指数函数、对数函数和幂函数，其最后一位可能与 Fortran 代码所用的 CPU 数学库（glibc）有所不同。因此，精确路径会在 CPU 上使用该数学库计算这些函数，再将结果返回 GPU。由于 PyTorch 无法对外部库调用进行求导，我们会以公式的形式提供这些函数的导数，从而使精确路径仍可微分。查找表也以同样的方式构建，因为微小的表值差异会延续到后续的非线性计算中。精确路径用于与 Fortran 代码进行核对，十二小时的 Marshall 对比也采用此路径。快速路径则使用标准 GPU 算术和编译内核（即在 GPU 上运行的小型程序）求解相同的方程。预报计时、导数计时以及使用神经网络生成的预报均采用快速路径。

转换流程

人工智能编码代理编写了 WRF-Speed 代码，我们将所有代码都与编译后的 Fortran 代码进行了核对。我们为这项工作制定了一条规则：凡是与 Fortran 代码存在差异，无论差异多么微小，在受控测试证明并非如此之前，一律视为错误（参见“数值差异诊断”）。为了让对比通过，我们从未接受任何残留的微小差异。

编码智能体分五个阶段重建每种物理方案：(1) 确定配置所用的源代码和设置；(2) 构建编译后的 Fortran 测试用例，以全精度打印输入、中间值和输出；(3) 转换计算；(4) 比较中间物理量和最终输出；(5) 将全精度数值保存为回归测试，并在此后每次更改代码后重新运行。必须通过一个阶段，才能开始下一阶段。对于较长的例程，我们逐节进行比较。我们在三个层面检查一致性：单个例程（例如，一个用风输送物理量的平流项，或一种物理方案）、组件间耦合（即确保每个值都以正确的单位、在正确的时间传递），以及完整的时间步。第三个层面很重要，因为单个组件单独运行时可能匹配，但组装后的模型却可能出错（见结果）。

用于构建 WRF-Speed 的人工智能编码代理包括 Claude（Anthropic）[54], Codex（OpenAI）[55],GLM 5.2 和 5.3[56], Kimi K2.7 和 K3[57,58], Grok 4.5[59], Gemini（Google）[60], Gemma（Google）[61] 以及若干 DeepSeek[62,63] 和 Qwen[64] 模型。在作者指导下，同一批模型还协助准备稿件。作者设计了验证流程，指导了相关工作，并对报告的结果负责。

数值差异诊断

我们通过两类受控测试，逐一调查所有非零差异。第一类测试中，我们只更改一项编译器设置，重新构建 Fortran 代码，然后观察差异是否发生变化。融合乘加收缩就是一种这样的设置：编译器将乘法和加法合并为单个运算，只进行一次舍入，而不是两次。另一项设置是优化级别，我们会比较经过优化和未优化的构建版本。第二类测试中，我们向 WRF-Speed 提供由 Fortran 代码计算出的中间值，以此判断差异是否源自更早的计算。

为确定差异在一个时间步内最先出现的位置，我们在时间步的每个例程执行后保存整个计算域上的 Fortran 状态。我们使用与相应 Fortran 输入按位相同的输入来初始化 WRF-Speed，

并一次执行一个例程。比较每个例程执行后的输出，即可确定两段代码首次出现分歧的例程。

耦合比较

WRF-Speed 使用精确路径，在一块 NVIDIA L40 GPU 上以双精度推进耦合模拟。Fortran 代码和 WRF-Speed 分别从各自的状态出发，使用相同的初始数据和侧边界数据。我们每两个模拟分钟比较一次保存的状态，共检查 365 个检查点。在每个检查点，我们比较每个计算域中的八个大气场：三个速度分量、位温、位势、柱质量、水汽和气压。对于火灾，比较内容包括火灾水平集函数、点火时间、燃料比例、燃烧比例、热通量和水汽通量、燃料燃烧时间常数、燃料类别，以及每个火灾网格单元中记录的三个火星计数器：生成的火星数量、着陆的火星数量和飞火点燃次数。

在每个检查点，我们比较 68,064,683 个值。其中，62,154,923 个浮点值通过其原始 64 位模式进行比较，因此即使 $+0$和 -0也会被视为不同。其余 5,909,760 个值是燃料类别和三个火星计数器中的两个。由于这两种代码中的存储类型不同，我们对这些值进行数值比较。第三个计数器在两种代码中的存储类型相同，因此纳入逐位比较。

在耦合运行中，我们根据仍在空中的火星的物理属性进行比较。比较时不考虑每个粒子的存放位置，也就是它在粒子数组中的位置，以及在 Fortran 代码中由哪个 MPI 进程持有（进程是 CPU 集群上分担工作的多个进程之一）。两个属性完全相同的火星仍计为两个，因此粒子数量也必须一致。

计时

我们对两种计算进行计时：一个预报步，以及一个预报步后接计算导数的反向传播。对于预报，一次样本包括 1 千米区域的一个 3秒步长，以及其中包含的九个嵌套子步。两种代码都运行马歇尔模拟中相同的 100 个步（2021年12月31日协调世界时 03:01:06–03:06:06），其中五个步包含辐射更新。单精度 Fortran 构建版本是用于生产环境马歇尔山火模拟的可执行文件[27]。双精度构建版本采用 WRF 的同一 GNU 编译器配置及其默认优化设置。它与用于逐位比较的可执行文件不同。两者都在一个 Derecho节点上使用 128 个 MPI 进程运行。WRF-Speed 在一块 A100 GPU 上运行，并先推进 20 个不计时的步（预热），以排除一次性设置开销。两种代码都关闭输出文件。每种代码都从其先前运行保存的模型状态（重启状态）开始，因此两者涵盖相同的模拟时段，但起始状态并不相同。

为减少 CPU 向 GPU 发送工作所花费的时间，WRF-Speed 会记录一次某个时间步的 GPU 工作，并在后续时间步重放该记录（即 CUDA 图，见补充材料第 S1 节）。执行辐射更新的时间步还会进行记录中未包含的额外计算，因此会按正常方式运行。为测试一项优化，我们进行了四次运行，每次运行 100 个时间步，并报告采用该优化的两次运行的每步平均时间（见补充材料第 S1 节）。每次运行结束时的模型状态，在位级别上都与此前一次未使用重放功能的 WRF-Speed 运行结果完全相同。补充表 S5 列出了计时结果。

对于导数计时，111 米基准测试会推进九个嵌套子步（模拟 3 秒），然后计算一个输出相对于两个输入的导数。输出是 t_m 在嵌套网格上的平均值； t_m 是 WRF 推进的温度变量（湿位温减去 300K）。输入是分别沿网格的x 方向添加到初始风场和该温度变量上的均匀偏移量。我们从 1 千米区域计算强迫，并

预先计算辐射加热率并将其固定。我们在 A100 上重复计算四次，在 GB200 上重复八次。

报告的时间为墙钟时间，在 GPU 完成工作后读取。它们不包括编译和一次性记录。每次重复运行时，模型状态、输出和导数都必须与未编译情况下运行的同一计算结果相同。这项检查确认，更快的执行得到了相同的结果。它并不检验导数本身；这正是下文梯度检查的目的。补充材料第 S1 节列出了各个样本和内存用量。

梯度检查

我们在三种情形下检验导数：1 千米 Marshall 区域的一个时间步、理想化地形上的火灾模型本身，以及 111 米 Marshall 嵌套区域的一个子步。在每种情形中，我们都将自动微分得到的导数与有限差分进行比较；后者通过稍微改变输入并重新运行模型获得。自动微分是 PyTorch 提供的一种功能，它会记录计算中的操作，并对这些操作应用链式法则。

对于 1 千米区域，我们计算 J 的导数，即一个模型时间步后 t_m 在整个计算域上的平均值。反向模式微分从输出开始反向计算，一次遍历即可得到 J 相对于每个输入的导数。前向模式微分则将选定输入的一个扰动沿模型向前传递。我们将未编译的 PyTorch 实现中通过反向模式微分得到的导数，与中心有限差分进行比较；后者根据模型在大小为 h 的正、负等幅扰动下的运行结果计算得出。对于三种扰动模式，我们使用从 10^{-1} 到 10^{-9} 的 h：网格风 U 的均匀变化、t_m 的均匀变化，以及平滑随机场。

我们依据三项判据评估导数：(1) 在连续三个扰动幅度下，与中心有限差分结果的相对差异至多为 10^{-6}；(2) 泰勒余项斜率为 2.0 ± 0.1；(3) 前向和反向模式的方向导数彼此相差不超过 10^{-12}。泰勒判据用于检验线性预测的 J 误差是否会随着扰动幅度的平方近似减小。第三项判据是点积检验。前向模式给出所选输入扰动引起的 J 变化，而反向模式导数与相同扰动相乘后必须得到相同的数值。

点积检验符合判据：在四个方向上，前向和反向模式的方向导数相差不超过 1.1×10^{-15}。相比之下，当模型过程会因扰动而开启或关闭时，有限差分比较不符合判据。例如，WSM6 微物理方案仅在云冰和雪的含量为正的位置计算沉积和沉降（$q_i > 0$, $q_s > 0$），而扰动可能改变满足这一条件的位置。出于相同原因，泰勒测试也不符合判据。

在强制条件语句保留未扰动运行中所选结果的情况下，我们重复进行了导数检查。在固定分支选择后，t_m 导数在连续四个扰动规模下与有限差分一致，相对差异为 2.7×10^{-10} 至 2.5×10^{-7}，对应 $h = 10^{-6}$ 至 10^{-9}。均匀-U 导数（-4.24×10^{-5} K（米 s^{-1}）$^{-1}$）在两个扰动规模下处于容差范围内；之后，J 的变化变得过小，无法在浮点运算中解析。平滑随机模式的导数也在两个扰动规模下保持一致。其一致性受到 YSU 边界层方案中切换的限制，而我们无法将这些切换固定。因此，这两项比较均未满足在连续三个扰动规模下保持一致的要求。

补充图 S1 中的火灾模型导数检查使用预设均匀风，不进行大气积分。模拟在 CPU 上以双精度运行，使用间距为 75 米的 68×68 火灾网格，时间步为 1 秒。我们将一座高度为 1,200 米、半径为 2,250 米的山地与平坦地形对照进行比较，并使用半径为 300 米的点火圆盘。

我们评估两个区域在固定时间之前释放的热量：以山脊顶部为中心、半径为 750 米的圆盘区域，以及距点火点下风向至少 750 米的区域。我们在五个运行点进行评估，每个运行点由风速、风向和燃料湿度组合而成（例如，沿上坡方向以 5 米/秒的风速吹风，湿度为 6%）。针对一个运行点、地形、输入和区域计算的每个导数，构成一个条目。由此得到 20 个下风向条目和 18 个山脊顶部条目，因为在平坦地形上，横坡情形在山脊圆盘区域内不会释放热量。输入发生扰动时，区域位置保持不变。上坡风沿 $+x$方向吹，横坡风沿 $+y$方向吹。我们计算相对于均匀风速和枯死燃料湿度的导数。

我们使用从 10^{-9} 到 10^{-3}的 13 个扰动大小，将自动导数 D_{AD} 与中心有限差分导数 D_{FD} 进行比较。当连续三个扰动大小满足以下条件时，导数符合接受准则

$$|D_{\mathrm{AD}} - D_{\mathrm{FD}}| \le 10^{-4} + 10^{-3}|D_{\mathrm{FD}}|.$$

对于风导数，绝对容差的单位为 GJ 每米每秒$^{-1}$；对于水分导数，单位为 GJ 每千克每千克$^{-1}$。其中，Q 是所选区域中的热释放量，x 是输入量（风速或燃料湿度），h 是扰动大小。当 $|Q(x+h) - Q(x-h)| < 10^{-12}|Q|$时，$Q$的变化在浮点运算中太小而无法分辨，因此我们不将该比较计为通过或失败。

全部 38 个条目均符合该判据，尽管个别扰动大小可能失败。火灾网格单元中燃烧单元数量的变化并不能解释这些失败。在 47 次失败的比较中，每一步的燃烧单元数量均未改变；而在 55 次成功的比较中，燃烧单元数量发生了变化。在执行 530 次前向模拟之前，我们修正了测试流程。

对于耦合的 111-m 导数检查，我们从模拟开始后 144 分钟时的状态启动 Marshall嵌套区域，并在固定区域的最低三个模式层上扰动网格-x 风。我们在启用大气物理、火势蔓延、热反馈和火星飞散的情况下，计算一个 1/3-s 子步长后的投影燃烧面积的反向模式导数。我们使用 0.01、0.03、0.1、0.3 和 1 m s^{-1} 的风扰动，将该导数与中心差分进行比较。容差为 10^{-5} acre (ms^{-1})$^{-1} + 10^{-3}|D_{\mathrm{FD}}|$。这些增量均未达到容差判据。记录的过火面积、水平集函数、燃料和飞火点火计数均未发生变化，但最大的扰动增加了两个活跃火星。

差异有两个来源。首先，火灾水平集方案（五阶加权本质无振荡方案，WENO5）会根据火线移动的方向选择要使用的相邻网格单元，而扰动可能会改变这一选择。其次，代码会限制风速，因此在该限制生效的位置，扰动不会产生影响。WENO5 计算与 Fortran 代码一致。当我们固定代码中的条件选择（即代码在备选方案之间进行选择的位置），使其保持未扰动运行中的状态时，在 0.01、0.03、0.1 和 0.3 m s^{-1} 的扰动下，导数达到容差（绝对误差为 3.3×10^{-9} 至 2.9×10^{-6}）。当允许条件选择发生变化时，对于 $^{-1}$ 及以下的 0.005 m s 扰动，导数达到容差。

火线校正实验

理想化耦合实验采用平坦的 2 千米计算域，其中包含间距为 50 米的 $40 \times 40 \times 50$ 个单元格、12.5 米的火灾网格和 0.25 秒的时间步，使用 Anderson 3类燃料（高草）以及三维 Smagorinsky湍流混合[65]。关闭辐射、微物理过程、陆面、边界层和火星飞散，同时开启火灾-大气耦合。对于火线校正实验，我们使用 WRF 4.8 未修改的 Fortran 代码，以 0.125 秒的时间步生成合成火线。我们使用不同的 WRF 版本和一半的时间步长，以确保观测数据并非来自我们所校正的配置。观测数据是在两个时间点沿 16 条固定射线测得的到火线的距离，

分别为 120 秒和 180 秒。我们添加标准差为 6.25 米、沿火线相关长度为 50 米的噪声。这些观测数据约束了两个初始风偏移量和初始燃料含水率。

对每个输入分别进行一次前向模式积分，共三次，从而在每个起始点得到一个 32×3 雅可比矩阵。该矩阵包含 32 个火线位置相对于三个输入的导数。对于两个时刻的每条射线，这些导数在连续三种扰动幅度下均与中心有限差分一致。通过反向模式计算的预报火线与观测火线之间失配的梯度，也通过了有限差分检验。高斯–牛顿更新利用雅可比矩阵估算能最大程度降低失配[11,41]的输入变化量。在运行任何校正后的预报之前，我们就已确定更新规则、更新最大幅度、输入范围和线搜索（补充材料第 S2.1 节）。

燃料含水率响应实验

含水率响应实验采用与火线校正实验相同的平地 2 千米几何结构和 0.25 秒大气时间步长，初始网格-x 风均匀，风速约为 3 米/秒$^{-1}$，并在线状点火期间（第 0–5 秒）进行点火。我们在有燃料网格单元上均匀改变初始燃料含水率，并将干燃料载量固定为 $D = 0.674/1.06$ 千克/米$^{-2}$。为保持干燃料载量不变，我们将燃料载量输入设为 $L = D(1 + m)$，其中 m 表示以千克/千克为单位的燃料含水率$^{-1}$。

我们沿着 $y = 843.75$、918.75、993.75、1068.75 和 1143.75m 处的五条固定横断线，对水平集的零等值线进行线性插值，以确定火头位置。这些位置的平均值定义为平均火头位置 $X(t)$。我们将 9 s 至时间 T 之间的平均蔓延速率计算为 $[X(T) - X(9)]/(T - 9)$。对该表达式求燃料湿度的导数时，我们将 $X(T)$ 和 $X(9)$ 对水分的依赖性都纳入计算。

我们使用绝对容差和相对容差的组合，将火头位置和平均蔓延速率的导数与中心有限差分进行比较。相对容差为 10^{-3}。火头位置的绝对容差为每 kg kg^{-1} 10^{-3} m，蔓延速率的绝对容差为每 kg kg^{-1} 10^{-5} (m s^{-1})。在 12 s 和 30 s 时，连续三个含水率扰动值 10^{-5}、3×10^{-6} 和 10^{-6} kg kg^{-1} 下的两个导数均满足容差要求。在 60 s 时，相应的扰动值为 3×10^{-6}、10^{-6} 和 3×10^{-7}。随后，我们将 6%、7%、9% 和 10% 水分条件下的新模拟结果与使用 8% 水分条件下计算出的导数所作的预测进行比较。在此之前，我们重新进行 8% 条件下的运行，并确认其结果与保存的运行结果完全一致。

诊断计算与图件制作

我们通过对火灾网格上的模型的燃烧比例进行积分来计算过火面积。为比较物理预报与神经网络预报，我们计算交并比，即两种预报中均被烧毁的面积除以任一种预报中被烧毁的面积。该计算使用由 LFN < 0 定义的二值燃烧面积掩码，其中 LFN 是火灾水平集函数。掩码采用原始火灾网格坐标，并省略额外且未使用的存储行和列。火线校正图使用相应模拟中保存的水平集场，不进行平滑或重采样。

计时图展示了 GPU 完成工作后读取的墙钟时间，以及分别在 GPU 本身上测得的前向与反向部分的时间。我们保留了所有计时样本，包括一次耗时 17.86 秒的 A100 重复实验，而其他三次耗时为 13.73 至 13.74 秒。补充材料第 S3 节给出了火灾网格坐标、燃烧覆盖范围的两种定义以及有关图的说明。

神经网络组件训练与耦合评估

我们训练了三个神经网络，分别用于短波辐射、长波辐射和火蔓延速率。训练目标是网络所替代的物理方案的输出。随后，我们在耦合模型中运行这些网络，并测量模拟大气和火灾的变化。

辐射训练使用生成的大气廓线，以及科罗拉多州北部 2020 年 Cameron Peak 火灾模拟中的列。我们选择了另一场火灾，以便 Marshall 预报能够在主要训练案例之外检验辐射网络。长波训练数据包括低温下的晴空廓线和多云廓线。我们按日期和空间组别将 Cameron Peak 的列划分为训练列和开发列，并留出开发列用于检验拟合效果。我们仅使用训练列估算均值和标准差，并据此缩放输入和输出。长波网络是独立测试的例外。我们使用所选 Marshall 列的物理方案输出进一步训练该网络，并再次将数据分为训练列和开发列，因此其 Marshall 预报会复用部分训练案例。

长波网络使用三个隐藏层，每层有 512 个单元，并采用 SiLU 激活函数。短波网络在每个垂直层都使用相同的两个 32 单元 SiLU 层，并预测入射辐射中被吸收、散射或透射的比例，这些比例之和为一。另一个独立的火势蔓延网络使用两个隐藏层，每层有 64 个单元，并采用 tanh 激活函数。蔓延网络的输出由两个隐藏层给出的非线性预测值与输入的直接加权和相加得到。该网络有 4,757 个可训练参数，并学习燃料属性、风和坡度改变蔓延速率所对应的速率因子。物理方案仍负责准备燃料和水分输入、设定蔓延方向、限制蔓延速率，并处理山地灌丛（灌木）燃料；这类燃料使用单独的公式。补充材料S4节说明了输入、输出、保留的计算、开发数据集和误差。

匹配的物理预报和神经网络预报从相同的初始条件和已存储的随机数状态开始。随后，每个预报都从各自的检查点独立推进。我们比较模式网格上的火灾几何形态、积分燃烧比例和大气差异，以量化累积的耦合效应。蔓延网络通过未合并版本的水平集代码运行，而物理方案使用一个更快的内核，将其若干操作合并执行。因此，这些比较同时包含网络及其执行方法所产生的影响。

额外的预报用于量化对微小数值扰动的敏感度：物理模型运行十次，五种网络配置各运行十次，另有四次物理预报用于影响范围和大气比较。每次受扰动的物理预报均保留与未受扰动预报相同的物理方案、强迫和随机数状态，并将初始风场改变一个末位单位。这些预报通过展示物理模型对舍入尺度扰动的响应程度，为解释网络导致的差异提供尺度参照。我们在共同的有效时刻比较过火面积、火灾几何形态和大气场（补充材料第 S4.3 和 S4.4 节）。补充表 S5 列出了每次运行所使用的硬件和算术路径。补充表 S1 列出了每项主要结果背后的配置和比较。

数据可用性

WRF 参考源代码可在 `https://github.com/wrf-model/WRF` 获取。Frediani 等人[27]描述了 Marshall 案例和火品牌参数化。他们的源代码可在 `https://github.com/NCAR/WRF-FirebrandSpotting` 获取。初始文件和边界文件、重启状态、配置，以及图和表背后的数据，均可在同行评审期间提供给编辑和审稿人。我们将在发表时将其存档至 Zenodo，并附上 DOI。

代码可用性

WRF-Speed（wrf-torch，版本 0.1.0）由 `https://github.com/NCAR/WRF-torch` 开发。源代码、测试、对 Fortran 代码的校正以及训练网络，均可在同行评审期间提供给编辑和审稿人。发表后，我们将根据 Apache-2.0许可证发布这些内容，并将其连同复现结果所需的软件版本、输入和操作流程一并存入 Zenodo，同时附上 DOI。

致谢

本研究得到美国国家科学基金会国家大气研究中心的支持；该中心是美国国家科学基金会根据合作协议编号1852977资助的主要设施。我们感谢计算与信息系统实验室（NCAR）提供Derecho[66]和Casper（`https://ncar.pub/casper`）的高性能计算支持，该实验室由美国国家科学基金会资助。我们还感谢德克萨斯大学奥斯汀分校的德克萨斯高级计算中心（TACC）提供Horizon系统的使用权限，使我们能够对NVIDIA GB200图形处理器上的性能进行基准测试。

作者贡献

JSS主导了研究、软件开发、数值实验和稿件准备。IL和JD参与了模型和软件设计，并提供了代码示例。MF提供了有关马歇尔山火模拟及其Fortran预言程序的WRF-Fire详情，并协助准备稿件。PJ协助设计了可微性示例。所有作者均修订了稿件。

利益冲突

作者声明不存在利益冲突。

补充信息

耦合天气与火灾模型的精确可微重写

目录

S1 重写模型的执行 **22**
S1.1 GPU 内核和 CUDA图.22 S1.2 前向计时.24 S1.3 微分执行.24 S1.4 计时样本、内存和软件.25

S2 导数测试和前沿校正 25
S2.1 合成火线校正方案.26

S3 火灾网格数组和图注 **27**

补充图和表 28

S4 辐射与蔓延网络 31
S4.1 训练数据.31 S4.2 离线开发误差.33 S4.3 五个单独的 24 小时预报.33 S4.4 十成员集合和六小时比较.33

S5 边界层网络 **37**

S1 重写模型的执行

WRF-Speed 有两种运行方式。精确路径会在每个 PyTorch 操作被调用时执行该操作，并使用与 Fortran 代码相同的数学库。我们仅对这条路径声称逐位一致。快速路径使用标准 GPU 算术和编译内核（在 GPU 上运行的小程序），我们在大计算域上对其计时并进行微分。全文中，父区域是 1 千米外层域，嵌套网格是承载火灾的 111 米内层域。本节介绍快速路径的构建方式，以及我们如何测量其开销。我们将快速路径中的每项改动，都与未包含该改动的快速路径计算进行比较，而不是与精确路径或 Fortran 代码比较。补充表 S1 列出了论文中每项结果所依据的计算、比较对象及其局限性。

S1.1 GPU 内核和 CUDA 图

重写后的程序首先将每项 Fortran 计算表示为对整个数组（张量）执行的运算。相互独立的网格单元和列会一起计算，而相互依赖的步骤则保持原有顺序。这种形式可以在 GPU 上运行，但速度较慢。许多数组运算都会分别启动一个 GPU 内核（启动是指 CPU 发出一条运行内核的请求），并将中间结果写入内存，供下一项运算读取。

我们通过两种方式降低这一成本。首先，编译器会将兼容的数组运算组合成 GPU内核，包括反向传播中的运算。其次，我们为部分例程手动编写了 CUDA 内核，包括平流（由风进行的输运）、声学更新（处理声波的短子步）以及部分物理计算。每个手写内核都保留了

补充表 S1：各项报告结果所采用的计算方式及其比较对象。精确轨迹、GPU 计时、导数检查和神经网络预报均为独立计算。所有 GPU 计时、导数基准测试和神经网络预报均使用快速路径，因此它们与快速路径或未编译的 PyTorch 计算结果进行比较，而不是与 Fortran 代码进行比较。

结果	计算和配置	比较对象	限制
逐位一致 一致	嵌套 Marshall 计算采用 双精度， WRF-Speed 精确路径在一块 GPU 上	修正后的 WRF 4.5.1 Fortran 代码在 32 个 MPI 进程上以 365 十二小时内的检查点。 每段代码都从其 自身状态推进	适用于比较字段 以及此 Fortran 构建版本 code
正向速度	在一块 A100 上，快速路径运行 100 个父步（95 个未执行辐射更新，另有 5 个 执行了辐射更新）。运行四次， 平均值为 1.166 秒（其中两次 采用分组平流）。	Fortran 代码的墙钟时间 在 128 个 CPU 核心上，覆盖 相同的模型时间窗口	不同的硬件和 分别生成的初始 状态。启动、编译 和输出不计入
前向和 反向计算速度	九个 111米嵌套网格子步， 父级强迫和辐射 倾向保持固定。四次 A100 重复运行和八次 GB200 重复运行 每种配置。一个编译 步骤，适用于 1 千米区域	未编译的 PyTorch 计算的状态、损失和梯度 来自相同 代码版本，复现于 每次重复	表明更快的执行 保留了计算结果。这并不是 对某个 Marshall 预报敏感度进行的有限差分测试。 Marshall 预报敏感度。 A100 和 GB200 的运行 在代码、构建和 检查点放置
导数 检查、理想化	具有规定风场的火灾模型 在五个运行点上， 耦合水分响应和 前沿位置雅可比矩阵	中心有限差分位于 三个连续增量	水分和前沿测试 通过。在火灾模型测试中不 每项比较均符合 判据（方法）
导数 检查，Marshall 尺度	111米嵌套网格的一个子步 带有风扰动，以及一个 1 千米区域在三个 方向上的步长	中心有限差分	在 111 米处，预先指定的五个 未修改增量均失效。 四个增量通过， 分支选择保持不变 , and 较小的未修改增量 不超过 0.005m s^{-1} 也 通过。在 1 千米处只有一个 方向符合 三增量规则 , with 分支选择保持固定
前沿校正	理想化耦合区域，包含 来自原始模型的合成锋面 WRF 4.8 和两个起始点	留出数据中的锋面位置 时间为 300 s。均方根误差变化 从 51.57 米降至 8.57 米，并从 28.08 米降至 6.68 米	真实输入未能 恢复。未进行验证 与观测到的火灾进行对比
单个神经网络 预报	五个24小时马歇尔预报 采用短波、蔓延、两者 长波或三者均替换为 网络	处于该 配置中的物理预报，推进 自身运行，并有四种 扰动物理预报作为 一个尺度	这些特定网络的耦合 响应的单个示例 网络
十成员 集合	10个24小时预报 物理模型，以及对于每个 五种网络配置， 对初始风场进行末位比特变化 风	过火面积的平均值和范围 物理集合中	仅限过火面积。该 网络并未经过调优，以 匹配物理集合
边界层 （YSU）网络	一次24小时预报，包含 替代 YSU 部分的网络， 以及十组六小时配对	24小时的物理预报 以及六小时的物理 六小时集合	24小时神经网络和 物理预报使用不同的 代码版本，因此 比较无法单独区分 网络的影响。该 离线拟合效果较弱

算术表达式及其顺序。如果输入不符合预期形式，它会退回使用原始 PyTorch 代码（或因误差而停止）。例如，WSM6 微物理方案的融合内核无需将临时数组写入内存，而在每个时间步的三个阶段中，前两个阶段对水分物种的平流调用会合并为一次启动。两个版本都使用双精度。我们会将每项此类更改与未做更改的快速路径进行比较。与 Fortran 代码的比较则使用精确路径。

CUDA 图会改变向 GPU 发送内核的方式。我们只需记录一次一个时间步的 GPU 工作，之后便可通过一次启动重放，因此 CPU 不再逐个发送内核。对于预报计时，一个计算图涵盖一个父区域步和九个嵌套子步。预报仍逐步推进，因为每次重放都从前一步生成的状态开始。重新计算辐射的步骤包含未记录在序列中的计算，因此会按常规方式运行，不进行重放。这种对持续推进的预报进行重放的方式仅支持前向运行，且仅限双精度、单个 GPU、最多两个计算域，并且不支持从嵌套网格向父区域反馈。嵌套网格内火灾向大气的反馈仍保持启用。四次计时运行中均未拒绝重放。微分基准测试使用不同的计算图：它会记录一次前向传播和一次反向传播，并在每次重复运行之间重置为相同的初始状态。

S1.2 前向计时

方法部分描述了计时窗口。这里补充两点细节。首先，母域的 Fortran计时器已经包含嵌套网格，因此我们不再将嵌套网格计时器加到其中。其次，我们会等待 GPU 完成后再读取挂钟时间。计时不包括启动、编译、图记录和输出，并保留每个测量步骤，包括较慢的辐射步骤。

我们进行了四次计时运行，以测试一项优化：将所有水分变量（水汽、云和降水）的输运计算一起发送到 GPU，而不是一次发送一个变量。按顺序，这四次运行的分组设置分别为关闭、开启、开启和关闭，除此之外没有其他变化。四次单独运行的均值位于“数据可用性”部分所述的数据存档中。采用分组平流时，母域步骤的平均耗时为 1.166 秒（不进行辐射更新时为 1.118 秒，进行辐射更新时为 2.084 秒）。不进行分组时，平均值为 1.170 秒。因此，分组可节省约 0.3% 的时间。分组运行中的辐射步骤慢 0.6%，并且仍计入平均值。单独的性能分析显示，采用分组后 GPU 内核减少了 416 个。四次运行结束时保存的状态彼此按位相同，也与此前在此窗口运行的 WRF-Speed 结果按位相同。

S1.3 微分执行

111 米基准测试对九个嵌套子步进行微分。我们在初始网格-x风场和湿位温上添加均匀偏移量，并计算经过九个 1/3 秒步后，t_m 的网格平均值如何随每个偏移量变化（方法）。这些导数通过反向模式微分得出，这种方法在机器学习中称为反向传播。每次重复都从相同的初始状态重新开始。父级强迫和辐射趋势项（即辐射方案加入模型状态的加热率）保持固定。单独的 1千米基准测试将一个步、其目标函数及其导数一并编译，并测得 GPU 时间中位数为 167.7 毫秒（补充表 S5）。

反向模式微分需要前向运行中的中间状态。为限制内存占用，我们只在计算区域的边界处保留中间数组，并在反向传播期间重新计算每个区域内的操作。在 A100 上，这些数组保存在 CPU 内存中，除 GPU 内存外还占用 20.1 吉比字节；CPU 与 GPU 之间双向传输所需的时间也计入测量时间。在 GB200 上，这些数组保留在 GPU 上。我们将所有手写和编译生成的导数代码与原始代码的 PyTorch 自动微分结果进行核对。我们还单独测试导数是否与有限差分结果一致（方法，梯度检查）。两种 GB200 配置仅在两个执行选项上有所不同，而且每次重复运行的状态、损失和梯度都逐位一致。

A100 和 GB200 的运行在代码版本、软件构建和检查点放置方面也有所不同，硬件也不相同，因此两者的运行时间并非纯粹的硬件对比。

S1.4 计时样本、内存和软件

前向计时是 200 个父步的平均墙钟时间，其中 100 个来自采用分组平流的两次运行中的每次运行。111 米配置的前向和反向计算时间，是 A100 上四次重复运行以及 GB200 上每种配置八次重复运行的墙钟时间中位数。我们仅在 GPU 完成工作后读取墙钟时间。前向和反向阶段时间是在 GPU 本身上计时的中位数，不包括重复运行之间的重置和检查。在 GB200 上，我们在与总墙钟时间不同的单独运行中测量阶段时间。所有计时均不包括启动、编译、图记录和预热。前向时间窗口包括新的辐射计算，而微分基准测试不包括这些计算。

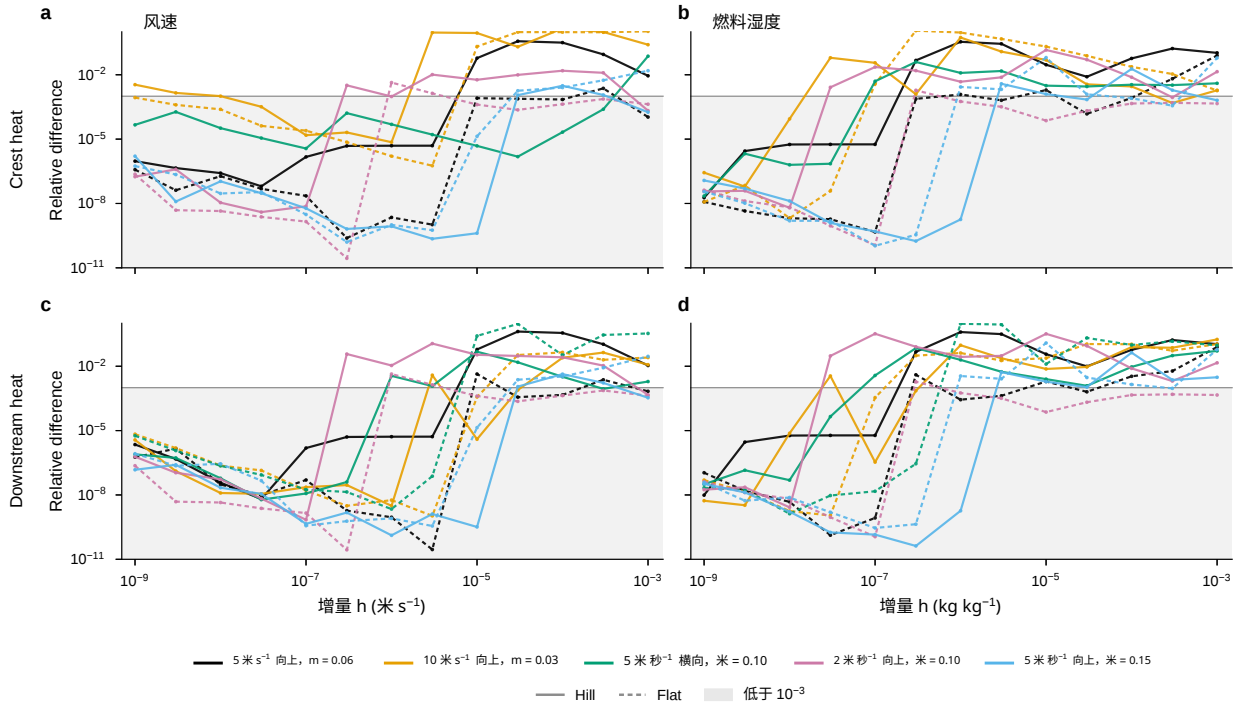
Fortran 计时使用的是 Derecho节点上的 WRF 源代码修订版 `cde4a22a`。该节点配备 128 个 AMD 处理器核心，每个核心使用一个消息传递接口进程。我们使用 GNU Fortran 12.2.0 和 Cray MPICH 构建了双精度可执行文件，并采用 WRF 针对分布式内存运行的标准 GNU 配置。实数被提升为八字节（`-fdefault-real-8 -fdefault-double-8`），优化标志采用该配置的默认值（`-O2 -ftree-vectorize -funroll-loops`）。128 个消息传递接口进程按 8×16 排列。单精度可执行文件是用于生产环境马歇尔山火模拟的 GNU 构建版本[27]。

四次 A100 墙钟时间分别为 13.73、13.73、13.74 和 17.86 秒。我们保留全部四次结果，并报告其中位数 13.73 秒。在 17.86 秒的样本中，GPU 阶段耗时 13.73 秒，因此额外的时间并非用于 GPU 计算。

内存占用取决于所处阶段，因为记录计算图所需的内存多于重放计算图。在 A100 上，整个执行过程中的分配峰值为 62.5 吉比字节，记录计算图期间为 54.2 吉比字节，而重放期间接近 15.0 吉比字节。对于 111-m 行，补充表 S5 报告的是整个执行过程中的峰值，其中包括准备和记录阶段，而不是较小的重放需求。A100 基准测试使用 PyTorch 2.14.1 和 CUDA 12.6，GB200 基准测试使用 PyTorch 2.14.1 和 CUDA 13.0。每个计时样本、两种 GPU 在各阶段的内存占用，以及软件和执行设置，均收录于数据可用性部分所述的数据存档中。

S2 导数测试与火线校正

方法部分介绍了我们如何将导数与有限差分进行比较：通过稍微改变输入重新运行模型，再用输出的变化量除以输入的变化量。补充图 S1 展示了火灾模型测试，其中涵盖了山地和平地上五个运行点处相对于风和燃料湿度的导数。山脊圆盘以山地上的 (2,587.5, 2,587.5) 米为中心，点火圆盘位于 (1,087.5, 2,587.5) 米处。并非所有比较都符合判据，方法部分报告了未通过的项目。两项马歇尔尺度检验仅部分通过。在 111 米尺度下，五个预先指定的增量均未能在未修改的有限差分下符合判据。固定未扰动运行的分支选择后，其中四个增量符合判据，且未修改的增量在 0.005 米/秒及以下时也符合判据。在 1 千米尺度下，未修改的有限差分不符合判据；固定分支选择后，三个方向中仅有一个符合三增量规则。本节其余部分介绍合成火线修正的步骤。补充表 S2 列出了正文中湿度响应比较（图 5a–c）所涉及的位移。



补充图 S1：火灾模型中自动导数与有限差分的比较。相对于增量 h，冠部圆盘（a、b）和下游（c、d）热释放的相对差异，分别针对风（a、c）或水分（b、d）求导。颜色表示五个运行点，实线和虚线分别表示山地和平地地形。灰色带表示相对项（10^{-3}）。完整的接受准则见方法。

S2.1 合成火线校正协议

前沿校正会调整三个输入（两个风偏移量和燃料湿度），使火灾预报火线与火线的合成观测相匹配。本小节给出完整步骤。控制向量为 $c = (\Delta U,\ \Delta V,\ m)$，其中初始水平风偏移量的单位为米/秒，$^{-1}$ 且 m为燃料湿度，单位为千克/千克$^{-1}$。优化逐元素使用 $q = (c - c_0)/s$，相应单位下的 $c_0 =$ 为 (0，0，0.09)，而$s =$ 为 (0.5，0.5，0.02)。目标函数为

$$\mathcal{L}(q) = \frac{1}{64}\,[r(q) - y]^{\mathsf{T}} C^{-1} [r(q) - y],$$

其中，r 包含在 120 s 时的 16 个径向距离，随后是 180 s 时的 16 个径向距离。每个距离表示沿固定射线测量的距离，即该射线首次离开过火面积的位置。偏离初始控制量不会受到惩罚。固定协方差具有相互独立的时间块，并且在每个时间块内$C_{ij} = 6.25^2[0.95\exp(-d_{ij}/50) + 0.05\delta_{ij}]$ 米^2，其中，d_{ij}是射线i 与 j 之间沿火线测量的最短距离，单位为米。对于两个起始点，我们使用相同的合成噪声实现。

利用位置雅可比矩阵 $A = \partial r/\partial q$（通过有限差分进行核验，见方法），更新采用 $g = A^{\mathsf{T}} C^{-1}(r - y)/32$、$H = A^{\mathsf{T}} C^{-1} A/32$ 和 $\delta q = -(H + \lambda I)^{-1} g$，其中$\lambda = 10^{-3}$ max(max diag $H, 10^{-12}$)。这是高斯–牛顿曲率，而非精确海森矩阵。欧几里得步长范数上限为 3。物理试探控制量被投影到 $\Delta U, \Delta V \in [-1.5, 1.5]$m s^{-1}和 $m \in [0.04, 0.14]$范围内。重新进行正向计算，并依次尝试步长乘数 $1, 1/2, 1/4, 1/8, 1/16$，接受实际目标函数严格下降的第一个步长。停止阈值预先固定为$\mathcal{L} \le 0.665$，在接受一次更新后应用；每个起始点最多接受三次更新。两个起始点

补充表 S2：燃料湿度变化时火锋位置的预测变化与模拟变化。燃料湿度为 8% 时的预报作为基准，比较 60 秒后的平均火头位置变化。预测值来自单次 8% 模拟中计算出的导数；模拟值来自各湿度条件下重新进行的耦合模拟；误差为两者之差占模拟变化的百分比。

燃料湿度（%）	预测变化（米）	模拟变化（米）	误差（%）
6	2.29	2.73	16.0
7	1.15	1.23	7.1
9	−1.15	−1.11	3.6
10	−2.29	−1.95	17.7

均在完成一次完整步长后停止，步长范数上限和控制量边界均未发挥作用。

两个起始点分别为（-0.75，0.50，0.09）和（0.50，-0.50，0.06）。它们各自被接受的控制量为

$$(0.09065, -0.25982, 0.071547) \quad \text{and} \quad (-0.30470, 0.02455, 0.049112).$$

目标函数值分别从 11.0223 降至 0.6551，以及从 3.7599 降至 0.6482。生成控制量为（0，0，0.07）。两次校正得到的控制量不同，且都不等于生成控制量。风和水分的变化可以相互补偿，因此在未恢复真实输入的情况下仍能匹配火线。300 秒时的火线不纳入目标函数、线搜索和停止准则。

补充表 S3：一次接受的校正后 300 秒时的火锋位置均方根误差。以无噪声生成预测为基准，使用 32 条固定射线计算误差，单位为米。拟合观测值位于 120 秒和 180 秒。

起始点	初始	仅风	仅水分	联合
第一	51.57	11.16	31.79	8.57
第二	28.08	3.15	14.54	6.68

受限控制比较。我们分别仅使用两个风偏移量或仅使用水分重新计算一次更新，同时将非活动控制量固定在其初始值。对于活动控制量 I，步长为 $\delta q_I = -(H_{II} + \lambda I)^{-1} g_I$，并采用与上文相同的 λ、目标函数、观测、边界和步长规则。从第一个起始点出发，仅更新水分会达到水分的下界。从第二个起始点出发，两次更新均接受完整步长。风调整从两个起始点出发均产生较小的 300 秒误差（补充表 S3），但水分仍保持其初始值。从第二个起始点出发，在 300 秒时，风调整的表现也优于联合更新。受限更新所接受的控制量和目标函数值记录在数据可用性部分所述的数据存档中。

S3 火灾网格数组和图注

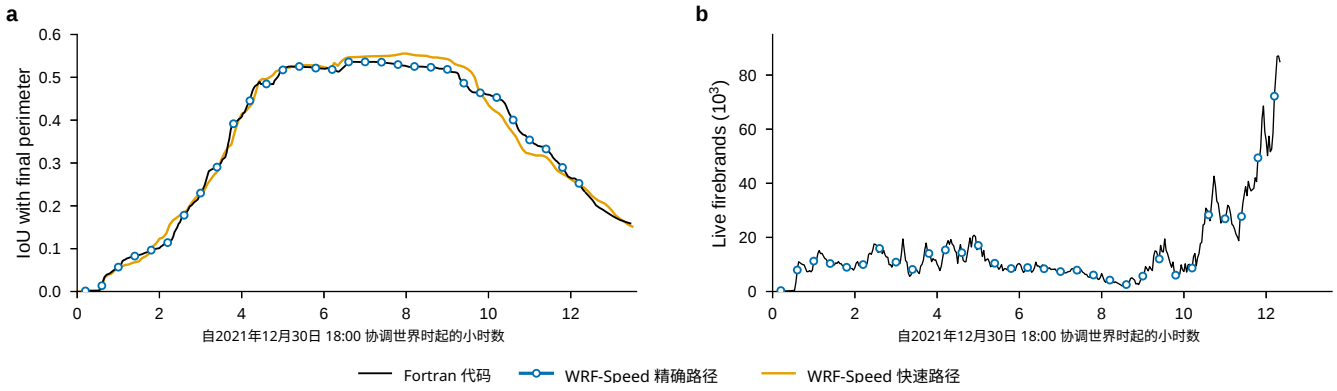
WRF 使用水平集函数追踪火灾，火灾已过的区域函数值为负。模型存储该函数时，额外增加一行和一列（1441×1369 个存储节点对应一个 1440×1368火灾网格）。模型不会更新这些节点；如果通过它们绘制等值线，就会在计算域边缘形成虚假的轮廓。因此，我们省略这行未使用的节点，并根据模型推进的节点及其原始坐标绘制火灾周界。每条等值线都

予以保留，包括零星火灾和未燃孔洞。过火面积是对所有火灾网格单元的燃烧比例乘以单元格面积后求和所得。

我们使用两种燃烧覆盖范围的定义。与观测周界的重叠部分（图 2 和补充图 S2）按燃烧比例非零的单元格计数。神经网络预报与物理预报之间的比较，则按水平集函数为负的单元格计数。观测周界是国家跨部门消防中心提供的最终火场边界（NIFC/WFIGS，6,079.90 英亩）。Fortran 代码大幅高估了观测到的最终面积，因此与其吻合只能说明计算过程得到保留，并不意味着预报与观测到的火灾相符。

前沿校正图（图 5）展示了保存的水平集场，未进行平滑或重采样。图 3 中的墙钟时间和阶段时间分别为 A100 四次重复运行或 GB200 八次重复运行的中位数，内存条显示各阶段的峰值。阶段时间在 GPU 上测量。GB200 的前向和反向阶段时间均来自与墙钟时间不同的一次运行。Fortran 前向成本汇总了两次各包含 100 个父步的重复运行；A100 前向成本则汇总了同一模型时间窗口内的两次分组平流运行。在补充图 S1 中，平地横坡情形的冠部圆盘释热为零，因此未予展示。

补充图表



补充图 S2：Marshall运行中的周界重叠和火星。a，对于 Fortran 代码（黑色）、WRF-Speed 精确路径（蓝色圆圈）和快速路径运行（橙色），与观测到的最终周界的交并比。b，活动火星；Fortran 代码与 WRF-Speed 精确路径的结果完全一致。

补充表 S4：为 Marshall 计算重写的组件。在将各组件组合运行之前，我们先逐一将其与 Fortran 代码进行核对。在 365 个检查点进行的耦合比较是对完整模型的测试。行数统计涵盖了包含所用例程的 WRF 源文件。统计的是整个文件，因此驱动程序文件还包括此配置未选择的方案分支。

组件	计算内容	文件	Fortran 代码行数
ARW 动力学	大气的风、气压和温度	12	50,707
单向嵌套	将 1 千米父区域的预报传递至 111米嵌套网格	5	9,091
YSU，修订版 MM5 地表层和 Noah	近地面混合，以及热量和水分交换与陆地	4	13,378
WSM6、RRTMG长波辐射和 Dudhia短波辐射	云和降水，以及红外辐射和太阳辐射	3	17,851
物理驱动程序和初始化	调用方案并应用其结果的代码	6	27,119
地表火灾水平集	火线如何蔓延以及释放多少热量	7	8,228
火星飞散	被风携带并引发新火灾的燃烧余烬	1	4,516
总计		38	130,890

补充表 S5：记录的模型执行成本。可微步骤对应的行使用预计算的辐射趋势项；111 米对应的行涵盖九个子步，且父级强迫保持固定。仅前向值为平均墙钟时间。可微步骤的前向值为在 GPU 上测得的中位数；111 米的前向和反向值为墙钟时间中位数；1 千米的前向和反向值为在 GPU 上测得的中位数。内存为整个运行期间的分配峰值。A100 的 111 米运行还使用了 20.1 吉比字节的 CPU 内存用于检查点。所有行均测量物理模型的运行时间，我们未报告神经网络带来的加速比。无意义，未测量。S1 节介绍路径和采样方法。

计算	前向	前向和反向	内存
可微步骤			
111米嵌套网格、九个子步、A100 80 GB	2.68 秒	13.73 秒	62.5 吉比字节
111米嵌套网格、九个子步、GB200、启动选项	1.36 秒	5.76 秒	114.1 吉比字节
111米嵌套网格，九个子步，GB200，默认启动	1.50 秒	6.39 秒	114.1 吉比字节
1 千米区域，每一步编译为一个程序，A100	52.1 毫秒	167.7 毫秒	n.m.
嵌套父步，仅前向			
Fortran 双精度，单个 CPU 节点	6.075 秒	–	–
Fortran 单精度（生产环境），单个 CPU 节点	2.769 秒	–	–
WRF-Speed 最终 CUDA 配置图重放，A100 80 GB	1.166 秒	–	n.m.

补充表 S6：用于耦合比较的 Fortran 代码校正，按其影响的计算分组。这些校正涉及非确定性、未定义行为、重启持久性、MPI 进程边界处的光环值以及一项诊断，其中一些会导致输出相对于未校正源代码发生变化。

受影响的计算	对 Fortran 代码所作的校正
粒子交换	在空进程上初始化接收掩码，使 MPI 缓冲区宽度与八字节实数匹配，并检查传入粒子容量。
持久粒子状态	在重启初始化期间保留活动火星。
火焰–空气交换	在边界处进行火灾风场插值之前更新粗糙度光环 MPI 进程之间。
未定义输出	在解除分配前读取生成上限诊断，并分配冰川分支上的冠层阻力。
最大距离	累积每个单元格中的最大沉积距离，而不是最后一次沉积距离。
重启输出	保留重复警报周期和待处理的首次历史输出
调度	时间。

S4 辐射与蔓延网络

我们训练了小型神经网络，用以替代三种物理方案，并将每个网络分别嵌入完整的耦合 Marshall 预报中运行。这样做的目的是展示如何插入和评估一个经过学习的组件，而不是建议将这些网络用于业务运行。这些网络分别替代 Dudhia 短波辐射方案[52]（阳光加热）、RRTMG 长波辐射方案（红外辐射造成的加热和冷却）以及 Rothermel 蔓延速率公式[43]（在给定燃料、风和坡度条件下火线的移动速度）。补充表 S8 说明了每个网络学习的内容，以及其周围保留了哪些物理计算。我们测试了五种组合：短波、蔓延、短波与蔓延、长波，以及三者。每个神经网络预报及其进行比较的物理预报，都从相同的 Marshall 初始状态开始，并使用相同的配置、输入文件、网格、时间步和随机数状态。随后，两者各自推进。所有预报均以双精度使用快速路径，因此这些预报与 Fortran 代码的逐位比较相互独立。预报于2021年12月30日协调世界时18:00开始，采用原始 Marshall 案例的初始条件和侧边界强迫，因此，6小时和24小时的结果分别对应12 月 31 日协调世界时00:00和18:00。指定点火的坐标和计时记录在“数据可用性”部分所述的数据存档中。

S4.1 训练数据

辐射训练采用来自科罗拉多州北部 2020 年 Cameron Peak 火灾（2020 年 8 月 13–15 日）的大气廓线、来自 2020 年 9 月 6–7 日的开发廓线，以及生成廓线。开发数据不用于训练，而是用于选择和评估网络。我们特意选择了这场不同的火灾，以便用 Marshall 预报检验辐射网络在主要训练案例之外的表现。

只有长波网络还使用取自物理 Marshall 预报的列进行训练，取样时间为预报起始后的 4、8、12、16 和 20 小时。这些数据从父区域和嵌套网格中共提供 16,640 列，并按保存时间和网格块（相邻列组成的区块）划分为 7,680 个训练列和 8,960 个开发列。8 小时和 20 小时的所有列均属于开发数据。每个训练批次抽取 52 个 Cameron Peak 列、26 个寒冷晴空列、25 个多云列和 25 个 Marshall 列。输入和目标值仅根据原始训练数据的统计量进行缩放，额外训练还会对偏离原始数据上先前拟合结果的情况施加惩罚。网络以 44 个模式层、45 个层界面和地表值为输入，并输出模式层上的通量和加热率。Cameron Peak 开发廓线全部为晴空。臭氧和温室气体含量固定为物理方案中的设定，云粒子尺寸不作为输入。

短波拟合使用了 90 个生成的训练族和 9 个 Cameron Peak 训练族，以及 100 个生成的开发族和 6 个 Cameron Peak 开发族。族是指一组相关的剖面，在训练或开发过程中会作为整体保留。只有两个 Cameron Peak 训练列包含云，Cameron Peak 开发列中则没有包含云的列，因此我们分别报告生成剖面和 Cameron Peak 的误差。

蔓延网络基于来自 64 个输入族的 32,768 次物理蔓延速率公式评估进行训练，并使用来自另外 16 个族的 8,192 次评估进行开发。训练涵盖 Anderson 燃料类别中的 12 类。类别 4（山地灌丛）在 WRF-Fire 中遵循单独的公式，我们予以保留。这五个输入是堆积比、风指数及其系数、风速和坡度的变换值。目标值来自物理公式，而非观测到的火灾周界。补充表 S7 列出了训练设置。

补充表 S7：辐射与蔓延网络的训练设置。数值来自用于预报的网络所保存的训练记录。短波网络分两个阶段训练。长波网络经过多个阶段的训练；表中列出的是添加 Marshall 列之前的最后一个阶段，以及使用 Marshall 列进行微调的阶段。对于长波网络，一个轮次包含 36 次更新。

网络与阶段	优化器	学习率	批次	训练时长	选择
短波，第一阶段	Adam	10^{-3}	512	100 个训练轮次	最低开发误差，在以下各项之间等权平均
短波，第二阶段	Adam	3×10^{-4}	512	额外300个训练轮次	生成项和 Cameron 峰值数据相同判据
长波，最后阶段 Marshall 之前列	AdamW，无权重衰减	10^{-4}	128	43个训练轮次	最低开发集目标函数。训练停止连续 15 个训练轮次未有改善
长波，使用以下内容进行微调 Marshall 列	AdamW，不使用权重衰减	10^{-4}，然后 10^{-5}	128	30 轮次，然后 85 训练轮次	最低 Marshall地表通量符合条件的轮次之间的均方根误差满足这三项 S4.2节中的条件。训练在连续 30 轮次未见改进后停止
蔓延	Adam，采用余弦学习率调度	2×10^{-3}，降至 2×10^{-5}	2,048	15,000 次更新	最低开发误差为三个对数响应

S4.2 离线开发误差

补充表 S8 列出了每个网络相对于其所替代物理方案的误差，这些误差均在耦合模型之外单独进行评估。所有评分均基于开发数据。短波网络和蔓延网络都紧密遵循其物理方案，对于每个物理量，R^2（方差解释比例，完美拟合时为 1）均高于 0.99。三者中，长波网络的准确性最低。

预报中使用的长波网络包括针对 Marshall 列的额外训练。因此，其 Marshall 评分并非独立测试，尽管 Marshall 开发列在训练时被留出。我们在满足以下三个条件的情况下选定该网络：每个原始训练组的误差均处于容差范围内，Cameron Peak 加热均方根误差低于 1.1 K day^{-1}，且冷晴空表面通量偏差处于 3 W m^{-2} 以内。仅长波预报和三者全用预报使用相同的长波网络。对于蔓延网络，我们仅报告开发评分，未评估单独的未处理测试集。蔓延评分针对完整的蔓延计算，其中水分缩放和方向计算仍采用物理方案。

S4.3 五个单独的 24 小时预报

本节中的五个单独预报即图 6a–c 所示的预报，我们将其作为示例展示。每个预报都是一个大型集合中的单个成员。因此，它与物理预报之间的差异，在大小和方向上都可能不同于正文所报告的集合平均差异；我们依据的是后者（S4.4 节）。这五个预报均在启用火灾–大气反馈和火星的情况下稳定运行了 24 小时，并且都与同一个物理预报进行比较（补充表 S9）。足迹交并比根据水平集函数为负值的单元格计算（S3 节），模型火场既未平移，也未插值。

为衡量这些单独预报的尺度，我们运行了四种扰动的物理预报。每种预报都将初始风场改变一个末位单位，运行 24 小时，并与未扰动物理预报进行比较。这组四个成员不同于 S4.4 节的十成员集合。在 24 小时时，四个成员的数值范围为 211.57–333.05 km^2,，而未扰动预报为 304.00 km^2；它们与该预报的足迹交并比介于 0.647–0.778 之间。在过火面积和足迹重叠方面，五个神经网络预报与物理预报之间的差异，均小于四种扰动预报中差异最大的那一个。

大气场的差异更大。补充表 S9 的大气列给出了 24小时 时嵌套网格完整存储场的均方根差，其中交错风已取平均并映射到网格单元中心。短波、长波和三者全用预报在每一列的差异都超出四成员范围。短波–扩展预报在地表温度、感热通量和两个向下辐射通量上的差异超出该范围，而其风差异则在范围之内。仅扩展预报在任何一列的差异都没有超出该范围。辐射网络会直接改变辐射加热和表面通量，因此，包含辐射网络的四种预报出现较大的大气差异是意料之中的。

S4.4 十成员集合与六小时比较

我们针对六种配置分别进行了十次 24小时 预报：物理预报，以及另外五种配置，其中网络分别取代短波方案、扩散方案、短波和扩散方案、长波方案，或上述三种方案。每个成员都将两个网格上的初始风分量改变一个末位单位，这是浮点数可能发生的最小变化。每个成员的预报时长都达到 24小时。表 S10 列出了每个集合中过火面积的平均值、最小值和最大值。

大多数成员落在物理集合的范围内，少数蔓延、长波以及三者兼有的成员位于该范围之外。这些网络仅用于演示，并未针对匹配该

补充表 S8：三个网络及其离线开发误差。对于每个网络，第一行说明所学习的内容、保留的物理计算，以及网络的输出阶段通过构造所强制执行的约束。参数数量包括学习得到的线性输出层。这些误差将每个网络与其所替代的物理方案进行比较，所用数据集为从训练数据中留出的开发数据。短波评分针对日照廓线。长波评分汇总每个数据集的所有列，长波加热是模式层上的温度加热率。RMSE 是均方根误差，倾向是方案添加到模型状态中的变化率。短横线表示未记录偏差。

开发数据	物理量	RMSE	Bias	R^2
短波网络（替代 Dudhia）。学习内容：每一层中阳光的吸收、散射和透射比例，来自一个由所有层共享的网络（$14 \to 32 \to 32 \to 3$、SiLU，1,635 个参数）。保留内容：入射阳光以及将吸收能量转换为加热率的过程。强制约束：比例之和为一，能量逐层向下传递，并且夜间加热率严格为零。				
生成廓线	位温倾向（K 天$^{-1}$）	0.1566	–	0.99819
	地表向下通量（W 米$^{-2}$）	6.681	–	0.99946
	整层吸收通量（W 米$^{-2}$）	1.828	–	0.99953
Cameron Peak	位温倾向（K 天$^{-1}$）	0.04088	–	0.99460
	地表向下通量（W 米$^{-2}$）	2.905	–	0.99992
	整层吸收通量（W 米$^{-2}$）	0.308	–	0.99962
长波网络（取代 RRTMG长波辐射）。学习内容：整层大气柱的辐射响应（639 个输入、3 个 512 单元 SiLU 隐藏层、378 个输出、1,046,906 个参数）。保留内容：物理方案的输入准备和固定气体含量。施加的约束：向上和向下通量均为非负值，并根据每层净通量的变化计算加热。训练时只有这个网络见过 Marshall 列，因此其 Marshall行并非独立测试。				
Cameron Peak（27,648 列）	地表向下通量（W 米$^{-2}$）	5.457	1.051	0.9219
	加热率（K 天$^{-1}$）	1.019	−0.102	0.9726
生成的冷晴空（256 列）	地表向下通量（W 米$^{-2}$）	1.642	0.227	0.9974
	加热率（K 天$^{-1}$）	0.302	−0.022	0.7388
生成的多云（1,792 列）	地表向下通量（W 米$^{-2}$）	8.442	1.463	0.9925
	加热率（K 天$^{-1}$）	1.422	−0.014	0.6019
Marshall（8,960 列）	地表向下通量（W 米$^{-2}$）	9.961	−2.608	0.8374
	加热率（K 天$^{-1}$）	1.694	0.048	0.7055
蔓延网络（取代 Rothermel 蔓延速率因子）。学习内容：无量纲基础因子、风因子和坡度因子的对数函数（$5 \to 64 \to 64 \to 3$、双曲正切函数，并通过从输入到输出的并行线性连接实现，共 4,757 个参数）。保留内容：燃料和含水量预处理、缩放至有量纲蔓延速率、投影至蔓延方向、速率上限、不可燃掩膜以及单独的灌丛公式。未对未处理测试集进行评估。				
16 个开发族	蔓延速率，完整计算（m s^{-1}）	0.001415	–	0.9999985

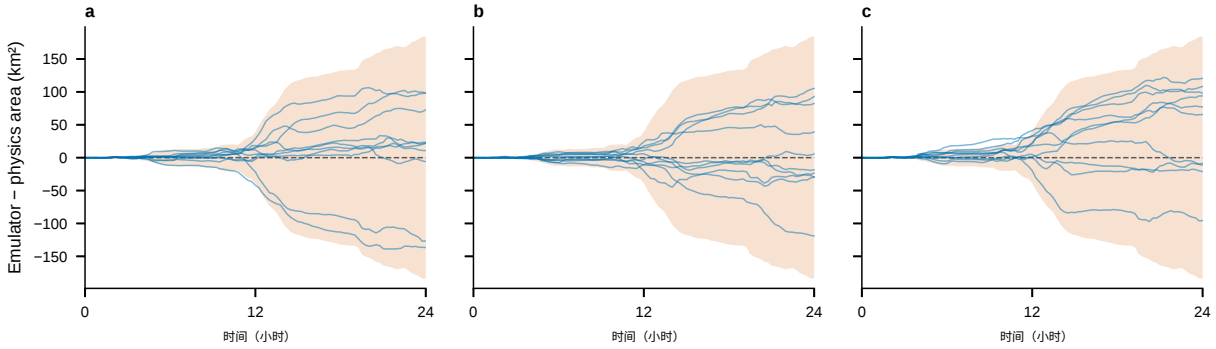
补充表 S9：采用网络的五个单独的 24 小时预报与物理预报的对比。面积变化和足迹交并比（神经网络与物理方案的烧毁区域的交并比）描述火灾。其余各列为 24 小时时刻 111米嵌套网格上的均方根差。风采用最低质量层， T_{skin} 为地表（皮肤）温度， H 为感热通量，SW↓ 和 LW↓ 分别为地表向下短波辐射和长波辐射通量，三者的单位均为 W m^{-2}。最后一行为四种扰动的物理量预报的范围，每个预报均与未扰动物理预报对比。其过火面积为 211.57–333.05 km^2, ，而未扰动物理预报为 304.00 km^2。所有数值均为单次预报的结果，仅作为示例。集合平均值见正文和补充表 S10。

	Fire		嵌套网格上的大气（均方根差异）				
网络	面积变化	IoU	风（米 秒$^{-1}$）	T_{skin} （K）	H	SW↓	LW↓
短波	−6.98%	0.777	1.84	1.32	22.24	62.83	18.87
蔓延	+8.40%	0.844	1.57	1.08	18.75	48.73	13.33
短波和蔓延	+1.77%	0.793	1.74	1.38	21.84	60.03	16.87
长波	+1.63%	0.711	1.85	1.48	22.84	54.39	23.24
三者	−21.53%	0.681	1.87	1.77	27.10	62.12	21.50
四种扰动的物理量	−30.4%至 +9.6%	0.647–0.778	1.73–1.81	1.12–1.18	17.60–20.20	44.21–53.15	12.77–15.77

补充表 S10：十成员集合在 24小时 时的过火面积。一个集合的成员之间仅在初始风场的末位比特变化上存在差异。最后一列是各集合平均值与物理集合平均值之比减一。

配置	Mean （千米²）	最小值 （千米²）	最大值 (千米²)	平均值之差 来自物理（%）
物理	264.3	187.5	371.5	–
短波网络	272.1	196.4	356.9	+2.9
蔓延网络	274.8	175.1	372.1	+4.0
短波和蔓延网络	307.0	264.7	357.2	+16.2
长波网络	333.3	212.4	416.1	+26.1
三者网络	288.8	191.2	385.0	+9.3

物理集合。正文中的百分比变化是集合平均值的比率。它们的大小和正负可能与 S4.3 节中的单次预报示例不同。图6显示了十成员范围。六小时的三十组成对预报中，短波、蔓延和短波–离散度各有十组，其平均过火面积相对于物理集合平均值分别相差 -0.44%、-0.28% 和 $+15.9\%$。另外，长波和边界层（YSU，S5节）的十组成对比较，其平均过火面积相对于同一物理集合分别相差 $+11.3\%$ 和 -8.4%。



补充图S3：成对集合响应。十对成员中，神经网络预报的过火面积减去具有相同初始扰动的物理预报过火面积所得的差值；分别展示短波（a）、蔓延（b）及二者的组合（c）。曲线采用共同的保存时刻，未进行插值。橙色带涵盖同一组十个物理成员之间的所有两两差值，±（最大值 A_{phys} − 分钟 A_{phys}），并作为数值敏感度的尺度。虚线表示零值。

S5 边界层网络

即使网络并不完美，仍然可以完成耦合预报；本节给出一个示例。YSU 方案用于计算大气最低部分——边界层中的湍流混合。它仅在 1 千米父域上运行。我们用两个神经网络替换了 YSU 中用于估算边界层高度、混合系数、非局地项（由大涡造成、并不遵循局地梯度的混合）和夹卷项（上方空气混入边界层的过程）的部分。每个网络均有三个隐藏层，每层包含 1,024 个单元，并具有 634 个缩放输入和有界输出。每一列中混合方程的数值解始终符合物理规律。预报过程中从未退回使用物理公式。

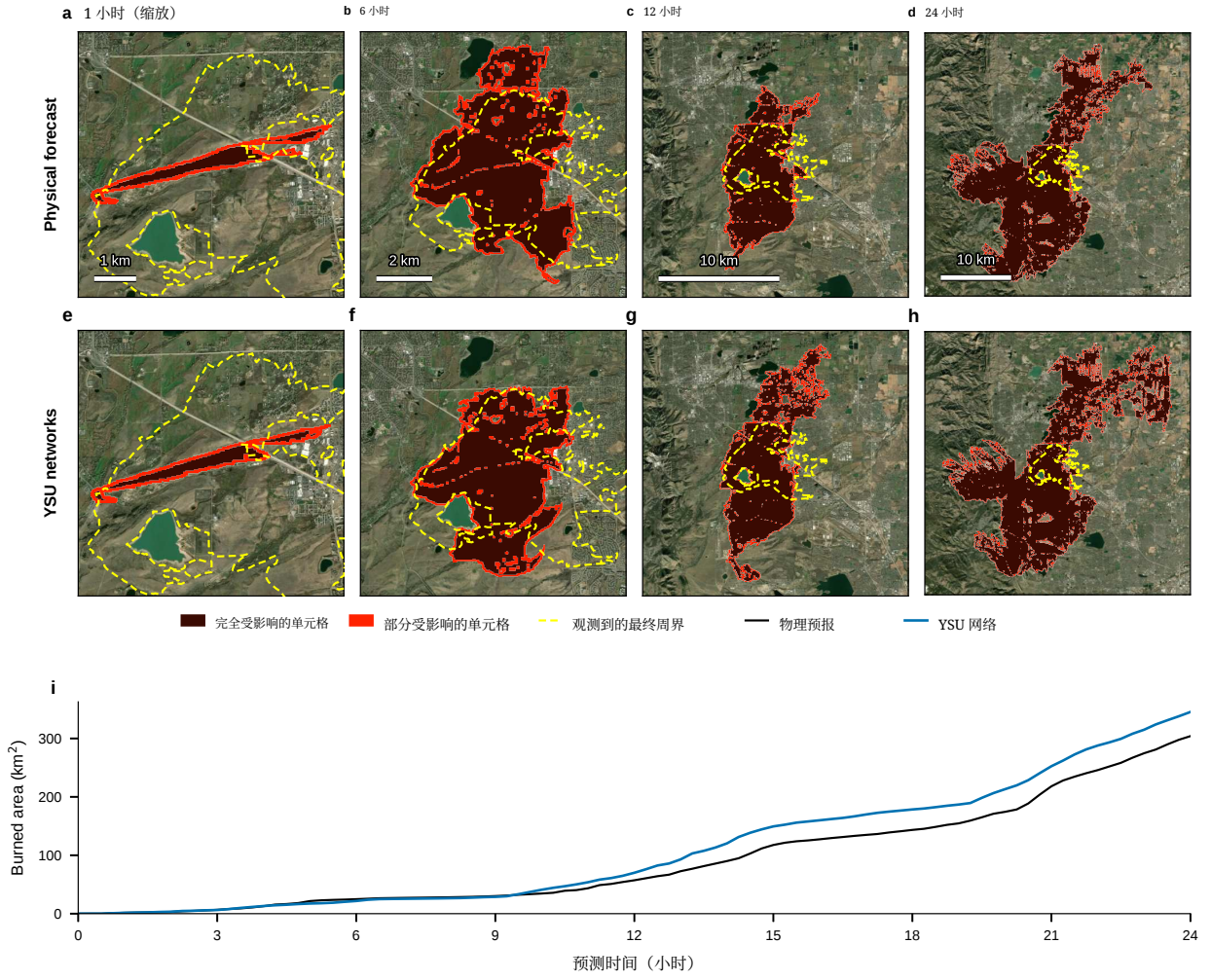
训练使用 Marshall 预报中 20 个保存时刻的数据，共 414,660 列（列索引的前 80%）。空间留出集是在这些时刻中不参与训练的列，即最后 10%（51,840 列）。中间的 10% 不进行评分。时间留出集包含另外九个时刻的全部 233,253 列。所有数据组均来自同一个火灾案例。我们仅使用训练列的统计数据对输入进行尺度缩放。首先训练网络 100 轮（遍历训练数据），使其再现所替代的量；然后再训练 5 轮，使其再现由此产生的倾向（方案添加到模型状态中的变化率）。表 S11 汇总了各列、各层和各保存时刻的误差。

补充表 S11：预报中使用的 YSU 网络的准确度。均方根误差以注明的物理单位表示。 $R^2 = 1 - \mathrm{SSE}/\mathrm{SST}$，其中 SSE 是误差平方和，SST 是每个组中物理方案值相对于其合并平均值的离差平方和。前六行是倾向（方案添加到模型状态中的变化率），并汇总全部 44 层的数据。边界层高度按列评估。

物理量（均方根误差单位）	训练 RMSE	空间留出集 R^2	空间留出集 均方根误差	时间留出集 R^2	时间留出集 均方根误差	R^2
温度（K 天$^{-1}$）	13.62	0.921	36.41	0.675	24.39	0.629
东风（米 s^{-1} day^{-1}）	72.00	0.977	141.09	0.923	249.31	0.733
北向风（米 s^{-1} day^{-1}）	31.86	0.962	66.67	0.922	81.22	0.803
水汽（g kg^{-1} day^{-1}）	5.24	0.774	10.40	0.387	10.30	0.264
云水 (克 千克$^{-1}$ 天$^{-1}$)	0.458	0.809	0.317	0.686	0.712	0.536
云冰 (克 千克$^{-1}$ 天$^{-1}$)	0.133	0.942	0.302	0.783	0.283	0.738
边界层高度 (米)	55.22	0.991	93.38	0.975	196.15	0.862

在每一步都调用网络且全程场值有限的情况下，预报完成了全部28,800个父步。在单次调用测试中，每列的热量、水汽和动量守恒误差均在 $3.05\epsilon S$以内，其中 ϵ 为机器精度， S 是列储存率和边界通量绝对值的总和。因此，列预算守恒且运行稳定，并不能保证倾向准确。留出时刻的水汽倾向拟合最差（$R^2 = 0.264$），温度和东风也并不完美（0。629和0。733）。

24小时火灾面积为345.37km^2, ，而物理预报为304.00km^2 ；足迹交并比（两个燃烧覆盖范围的交并比）为0.759（图S4）。网格坐标、燃料类别和保存时刻一致，但神经网络预报与物理预报采用了不同的代码版本，因此这项比较无法单独反映网络的影响。在24小时，五项大气差异均超出了四种扰动物理预报的范围（表S12）。另有一项独立的十对比较，对初始风场施加末位比特变化，并使用相同网络运行六小时；其平均过火面积比物理集合平均值低8.4%（S4.4节）。



补充图 S4：YSU 网络完成了 24 小时的 Marshall 预报。a–h，物理预报（a–d）和采用 YSU 网络的预报（e–h）在 1 小时（放大图）、6 小时、12 小时和 24 小时时的完全受影响（深红色，燃烧比例至少为 0.999）及部分受影响（亮红色）火灾网格单元。黄色虚线标示观测到的最终周界。i，物理预报（黑色）和采用 YSU 网络的预报（蓝色）的过火面积，每 15 分钟记录一次。图中显示了名义父步长时间。底图：Esri World Imagery。

补充表 S12：24小时采用 YSU 网络的预报与大气差异。YSU 列给出了采用 YSU 网络的预报相对于物理预报的均方根误差，分别针对父区域和嵌套网格。两种预报使用了不同的代码版本，因此这些差异无法单独反映网络的影响。物理范围是四种扰动的物理量预报中与未扰动物理预报相比的最小值和最大值。作为比较基础的两种物理预报在比较字段中按位相同。

	父区域		Nest	
物理量	YSU	物理范围	YSU	物理范围
风矢量（米 s^{-1}）	1.58	0.31–0.36	2.07	1.73–1.81
地表温度（K）	0.79	0.43–0.54	1.60	1.12–1.18
感热（W 米$^{-2}$）	12.93	8.22–10.95	29.01	17.60–20.20
向下短波辐射 (W 米$^{-2}$)	36.16	22.49–29.58	79.47	44.21–53.15
向下长波辐射 (W 米$^{-2}$)	11.70	6.28–8.54	23.57	12.77–15.77

参考

[1] Bi, K. 等. 利用三维神经网络实现准确的全球中期天气预报。《自然》619, 533–538 (2023).
[2] Lam, R. 等. 学习高技巧的全球中期天气预报。《科学》382, 1416–1421 (2023).[3] Price, I. 等. 利用机器学习进行概率天气预报。《自然》637, 84–90 (2025).[4] Bodnar, C. 等. 地球系统基础模型。《自然》641, 1180–1187 (2025).[5] Schreck, J. S. 等. 社区研究地球数字智能孪生：一个可扩展的人工智能驱动地球系统建模框架。npj《气候与大气科学》8, 239 (2025).[6] Chapman, W. E. 等. CAMulator：社区大气模式的快速模拟器。arXiv 预印本 arXiv:2504.06007 (2025).[7] Watt-Meyer, O. 等. ACE2：准确学习次季节至年代际大气变率及强迫响应。npj《气候与大气科学》8, 205 (2025).[8] Easterbrook, S. M. & Johns, T. C. 为理解气候变化而开发软件。《科学与工程计算》11, 65–74 (2009).[9] Schulthess, T. C. 等. 回顾百亿亿次级计算的目标与基准：基于天气与气候模拟的路线图。《科学与工程计算》21, 30–41 (2019).[10] Bauer, P. 等. 地球系统科学的数字革命。《自然·计算科学》1, 104–113 (2021).[11]Le Dimet, F.-X. & Talagrand, O. 气象观测分析与同化的变分算法：理论方面。Tellus A 38, 97–110 (1986). URL https://www.tandfonline.com/doi/abs/10.3402/tellusa.v38i2.11706.[12] Wang, X., Han, Y., Xue, W., Yang, G. & Zhang, G. J. 利用具有大气湿物理过程和辐射过程神经网络参数化方案的现实大气环流模式进行稳定气候模拟。《地球科学模型开发》15, 3923–3940 (2022).[13] Kochkov, D. 等. 用于天气与气候的神经网络大气环流模式。《自然》632, 1060–1066 (2024).[14] Paszke, A. 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（NeurIPS 2019），8024–8035 (2019).[15] Shen, C., Appling, A. P., Gentine, P. 等. 以可微建模统一地球科学中的机器学习与物理模型。《自然综述：地球与环境》4, 552–567 (2023). URL https://www.nature.com/articles/s43017-023-00450-9.观点文章。[16] Dahm, J. 等. Pace v0.2：基于 Python 的可移植性能大气模式。《地球科学模型开发》16, 2719–2736 (2023).[17] Rosinski, J. M. & Williamson, D. L. 全球大气模式中的舍入误差累积与移植验证。《SIAM科学计算杂志》18, 552–564 (1997).

[18] Baker, A. H. 等. 社区地球系统模型（pyCECT v1.0）的一种新型基于集合的一致性检验。《地球科学模型开发》8, 2829–2840 (2015).[19] Arteaga, A., Fuhrer, O. & Hoefler, T. 设计位级可复现的可移植高性能应用。载于2014年电气电子工程师学会第28届国际并行与分布式处理研讨会, 1235–1244 (2014).[20] Donahue, A. S. 等. 迈向百亿亿次级及更高：简洁的云解析 E3SM 大气模型（SCREAM），一个面向云解析尺度的可移植性能全球大气模型。《地球系统建模进展》16, e2024MS004314 (2024).[21] Zhong, X. 等. WRF–ML v1.0：连接 WRF v4.3 与机器学习参数化方案的桥梁及其在大气辐射传输中的应用。《地球科学模型开发》16, 199–209 (2023). URL https://gmd.copernicus.org/articles/16/199/2023/.[22] Skamarock, W. C. 等. 高级研究 WRF 模型第4版说明。NCAR 技术说明 NCAR/TN-556+STR，美国国家大气研究中心 (2019).[23] Powers, J. G. 等. 天气研究与预报模式：概述、系统工作与未来方向。《美国气象学会通报》98, 1717–1737 (2017).[24] Mandel, J., Beezley, J. D. & Kochanski, A. K. 利用 WRF 3.3 和 SFIRE 2011 进行大气与野火耦合建模。《地球科学模型开发》4, 591–610 (2011).[25] Coen, J. L. 等. WRF-fire：利用天气研究与预报模式进行天气与野火耦合建模。《应用气象与气候学杂志》52, 16–38 (2013).[26]西部水资源评估项目。复合灾害与马歇尔山火（2022年1月13日）。科罗拉多大学博尔德分校 (2022). URL https://wwa.colorado.edu/resources/hazard-assessments/briefing/compound-hazards-and-marshall-fire-january-13-2022.[27] Frediani, M. E. 等. 在 WRF-fire 中模拟火星飞散，以进行火灾与天气耦合预报。《火灾》8, 374 (2025).[28] Ranasinghe, N. R. 等. 将遗留 FORTRAN 代码通过大语言模型辅助翻译转换为 C++：一项跨平台研究 (2025). 预印本见 https://arxiv.org/abs/2504.15424.[29] Zhou, A., Hawkins, L. & Gentine, P. 概念验证：使用 ChatGPT 将地球系统模型从 Fortran 翻译并现代化为 Python/JAX (2024). 预印本见 https://arxiv.org/abs/2405.00018.[30] Lahlou, A., Hawkins, L. & Gentine, P. 将遗留科学代码系统化地通过大语言模型翻译为可微框架：在陆面模型中的应用 (2026). 预印本见 https://arxiv.org/abs/2606.07681v1.[31] Linck, I., Dennis, J., Schreck, J., Stengel, K. & Gagne, D. J., II. 评估大语言模型将遗留 Fortran 大气物理代码翻译为 GPU 加速的 JAX 的能力 (2026). 预印本见 https://arxiv.org/abs/2610.12662.[32] Zhang, X., Huang, X.-Y. & Pan, N. 天气研究与预报（WRF）模式升级版切线线性与伴随模式的开发。《大气与海洋技术杂志》30, 1180–1188 (2013).

[33] NCAR中尺度与微尺度气象实验室。《WRFDA用户指南》第6章：WRF资料同化，3.9.1版。 https://www2.mmm.ucar.edu/wrf/users/wrfda/Docs/user_guide_V3.9.1/users_guide_chap6.htm (2017)。[34] NCAR中尺度与微尺度气象实验室。《WRF用户指南：WRFDA概述》。 https://www2.mmm.ucar.edu/wrf/site/documentation/users_guide/overview.html (2026)。访问日期：2026年11月6日。[35] Davenport, E. H. 等。 JCM v1.1：一种可微的中等复杂度大气模型。《地球科学模型开发》19, 6451–6466 (2026)。网址 https://gmd.copernicus.org/articles/19/6451/2026/。[36] Yuan, S.等。 WRF-JAX v1.1.28：一种可微的区域天气模型，具有完全耦合的物理过程，与WRF在数值上一致，并在单个GPU上达到相当于128个CPU进程的性能（2026）。预印本见 https://arxiv.org/abs/2610.35095。[37] Tillet, P., Kung, H. T. & Cox, D. Triton：一种用于分块神经网络计算的中间语言和编译器。载于第*3*届*ACM SIGPLAN*机器学习与编程语言国际研讨会论文集（*MAPL 2019*），10–19（美国计算机协会，美国亚利桑那州凤凰城，2019）。[38] Ansel, J. 等。 PyTorch 2：通过动态Python字节码转换和图编译加速机器学习。载于第*29*届*ACM*编程语言与操作系统架构支持国际会议论文集，第*2*卷（*ASPLOS '24*），929–947（美国计算机协会，美国加利福尼亚州拉霍亚，2024）。[39] Mandel, J., Beezley, J. D., Coen, J. L. & Kim, M. 野外火灾资料同化：大气-地表耦合模型中的集合卡尔曼滤波器。《IEEE控制系统杂志》29, 47–65 (2009)。网址 https://ieeexplore.ieee.org/document/4939311/。[40] Mandel, J., Beezley, J. D., Kochanski, A. K., Kondratenko, V. Y. & Kim, M. 野外火灾-大气动态数据驱动应用系统中的火场边界数据同化及与燃料湿度的耦合。《计算机科学论文集》9, 1100–1109 (2012)。网址 https://www.sciencedirect.com/science/article/pii/S1877050912002402。[41] Rochoux, M. C., Ricci, S., Lucor, D., Cuenot, B. & Trouvé, A. 迈向野火蔓延的预测性数据驱动模拟——第一部分：基于多项式混沌替代模型、用于参数估计的低成本集合卡尔曼滤波器。《自然灾害与地球系统科学》14, 2951–2973 (2014)。网址 https://nhess.copernicus.org/articles/14/2951/2014/。[42] Rochoux, M. C., Emery, C., Ricci, S., Cuenot, B. & Trouvé, A. 迈向野火蔓延的预测性数据驱动模拟——第二部分：用于野火蔓延前沿追踪模拟器状态估计的集合卡尔曼滤波器。《自然灾害与地球系统科学》15, 1721–1739 (2015)。网址 https://nhess.copernicus.org/articles/15/1721/2015/。[43] Rothermel, R. C. 用于预测野外可燃物火势蔓延的数学模型。研究论文INT-115，美国农业部林务局，内陆山地森林与牧场试验站（1972）。[44] Goldberg, D. 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)。网址 https://dl.acm.org/doi/10.1145/103162.103163。

[45] Wicker, L. J. & Skamarock, W. C. 使用前向时间格式的弹性模型时间分裂方法。《月刊天气评论》130, 2088–2097 (2002)。[46] Park, S.-H., Klemp, J. B. & Kim, J.-H. WRF-ARW中的混合质量坐标及其对高层湍流预报的影响。《月刊天气评论》147, 971–985 (2019)。[47] Hong, S.-Y., Noh, Y. & Dudhia, J. 一种显式处理卷入过程的新型垂直扩散方案。《月刊天气评论》134, 2318–2341 (2006)。[48] Jiménez, P. A. 等。一种修订后的WRF近地层公式方案。《月刊天气评论》140, 898–918 (2012)。[49] Chen, F. & Dudhia, J. 将先进的陆面水文模型与宾夕法尼亚州立大学-美国国家大气研究中心MM5建模系统耦合。第一部分：模型实现与敏感度。《月刊天气评论》129, 569–585 (2001)。[50] Hong, S.-Y. & Lim, J.-O. J. WRF单矩六类微物理方案（WSM6）。《韩国气象学会志》42, 129–151 (2006)。[51] Iacono, M. J. 等。长寿命温室气体造成的辐射强迫：使用AER辐射传输模型进行计算。《地球物理研究杂志》113, D13103 (2008)。[52] Dudhia, J. 使用中尺度二维模型对冬季风试验期间观测到的对流进行数值研究。《大气科学杂志》46, 3077–3107 (1989)。[53] Anderson, H. E. 用于确定燃料模型以估算火行为的辅助方法。《通用技术报告INT-122》，美国农业部林务局，内陆山地森林与牧场试验站（1982）。[54] Anthropic。Claude。 https://www.anthropic.com/claude(2026)。[55] OpenAI。Codex。 https://openai.com/codex(2026)。[56] GLM-5-Team等。GLM-5：从氛围编程迈向智能体工程（2026）。网址 https://arxiv.org/abs/2602.15763。预印本见 https://arxiv.org/abs/2602.15763。[57] Team, K.等。Kimi K2：开放式智能体智能（2025）。网址 https://arxiv.org/abs/2507.20534。预印本见 https://arxiv.org/abs/2507.20534。[58] 月之暗面。Kimi K3。 https://www.moonshot.ai(2026)。于2026年7月16日发布。[59] xAI。Grok 4.5模型卡。 https://media.x.ai/v1/website/card-7f81d41b.pdf(2026)。[60] Gemini Team。Gemini 2.5：以高级推理、多模态、长上下文和新一代智能体能力推动前沿发展（2025）。网址 https://arxiv.org/abs/2507.06261。预印本见 https://arxiv.org/abs/2507.06261。[61] Gemma Team。Gemma 3技术报告（2025）。网址 https://arxiv.org/abs/2503.19786。预印本见 https://arxiv.org/abs/2503.19786。[62] DeepSeek-AI等。DeepSeek-V3技术报告（2024）。网址 https://arxiv.org/abs/2412.19437。预印本见 https://arxiv.org/abs/2412.19437。[63] Guo, D.等。DeepSeek-R1通过强化学习激励大语言模型进行推理。《自然》645, 633–638 (2025)。

[64] Yang, A.等。 Qwen3 技术报告（2025）。网址 https://arxiv.org/abs/2505.09388。预印本发表于 https://arxiv.org/abs/2505.09388.[65] Smagorinsky, J. 使用原始方程进行的大气环流实验：I. 基本实验。《月刊天气评论》91，99–164（1963）。网址 https://journals.ametsoc.org/doi/abs/10.1175/1520-0493%281963%29091%3C0099%3AGCEWTP%3E2.3.CO%3B2.[66] 计算与信息系统实验室。Derecho：HPE Cray EX 系统（NCAR 社区计算）（2023）。