arXiv:2603.02809v1 [cs.LG] 3 Mar 2026

基于格的深度神经网络：正则性与定制正则化

Alexander Keller、Frances Y. Kuo、Dirk Nuyens 和 Ian H. Sloan

2026年3月

摘要

本文综述了格点规则在深度神经网络（DNN）中的应用。格点规则是准蒙特卡罗方法的一类，已在高维积分和函数逼近等多种场景中展现出良好的效果。由于其形式极为简单，实现起来非常容易——只需一个长度与问题维数相匹配的优质整数生成向量。近年来，关于深度神经网络应用与理论的研究蓬勃发展。我们回顾了近期一篇文章，其中将格点规则用作具有光滑激活函数的深度神经网络的训练点，并得到了深度神经网络的显式正则性界。通过对网络参数施加限制，使其符合目标函数的正则性特征，我们证明，采用定制化格训练点的深度神经网络能够获得良好的理论泛化误差界，且隐含常数与输入维数无关。我们还通过数值实验表明，采用定制正则化训练的深度神经网络，其表现显著优于采用标准l_2 正则化训练的网络。

关键词： 格点规则、格算法、深度神经网络、定制化格训练点、定制正则化。

1 引言

本文综述了所谓格点规则在深度神经网络（DNN）[29, 61]中的应用。格点规则是一类准蒙特卡罗（QMC）方法，是用于近似 s维单位立方体上积分的等权求积规则

$$I(F) := \int_{[0,1]^s} F(\boldsymbol{y})\,\mathrm{d}\boldsymbol{y} \quad \approx \quad Q(F) := \frac{1}{N}\sum_{k=1}^{N} F(\boldsymbol{t}_k), \tag{1}$$

其中，点 $\boldsymbol{t}_1, \ldots\ldots, \boldsymbol{t}_N \in [0,1]^s$ 经过设计，比随机点分布得更均匀，旨在改善简单蒙特卡罗方法缓慢的 O（$N^{-1}/2$）收敛速度。庞大的准蒙特卡罗研究人员群体积极推动着这一领域的发展。该领域的专著和综述文章包括 [19, 21, 23, 38, 58, 62, 63, 74, 84]。本文综述是为该领域研究进展专题双年会议第16届会议论文集而撰写的。

我们（本文的四位作者）尤其偏爱格点规则，因为它们易于实现，而且其群结构有助于误差分析。格点规则（以及更广泛而言的准蒙特卡罗方法）在众多应用领域都取得了显著成效。本文作者的代表性贡献涵盖计算机图形学（例如 [15, 16, 45, 46, 49]）、具有随机输入的偏微分方程（PDE）的不确定性量化（例如 [20, 26, 31, 32, 33, 36, 37, 40, 41, 42, 55, 57]），以及数理金融（例如 [1, 2, 27, 88, 89, 90]）；准蒙特卡罗技术最初是在奠基性工作 [3, 10, 78] 中进行探索的。

在这篇综述文章中，我们总结了近期工作的研究发现 [47]。在该工作中，我们将深度神经网络视为用于函数逼近的非线性算法，并采用格点作为训练点。这些

基于格的深度神经网络因此被视为截断三角级数、多变量多项式和核方法等逼近算法的替代方案。一个自然的问题是，这些方法的性能孰优孰劣。我们提供了一些初步的理论见解，并鼓励进一步探索这一领域。准蒙特卡洛方法在机器学习中的其他应用 [34, 70, 82]包括诸如 [17, 25, 30, 48, 66, 67, 69, 91], 的研究，尤其是关于 [8, 64, 65] 变分推断的研究。

本文的结构如下。在第 2 节中，我们概述用于积分的格点规则的三种理论设定。此外，我们还强调了关于积分和逼近的相关结果，这些结果未在 [47] 中涵盖。在第 3 节中，我们首先介绍深度神经网络，并说明在进行误差分析时，需要对深度神经网络具有正则性界。我们总结了 [47] 中的三个关键定理：定理 1 根据“网络参数”和“激活函数”的导数界给出了深度神经网络的正则性界；定理 2 表明，我们可以限制深度神经网络的网络参数，使其匹配目标函数的正则性特征，从而控制影响“泛化差距”的范数；定理 3 声明，我们可以构造格点生成向量，在与输入维数无关的情况下达到某些误差界 s。我们还介绍了 [47] 中的一种定制正则化方法，数值结果表明，该方法在 [47] 中的表现优于标准l_2 正则化。在第4节中，我们进行数值实验，并在第 5 节中给出总结。

虽然本文主要是一篇综述，但也包含一些新的贡献。在引理1中，我们推广了 [47] 中激活函数的导数界。具体而言，我们考虑广义的“swish”激活函数 $x/(1+e^{-cx})$，当 $c\to\infty$ 时，它逐点收敛到“ReLU”激活函数 $\max\{x,0\}$。在引理2中，我们通过给出一个下界，证明 [47] 和引理1中得到的导数界的阶乘增长无法避免。第4节中对参数变化的广义 Swish函数所做的数值实验是新的。在附录中，我们给出定理 3 的完整证明细节，该证明在 [47] 中被省略；这些细节用到了第 2 节中的综述材料。

2 格理论在积分与逼近中的综述

（秩-1）格点规则的 N 个点定义为

$$\boldsymbol{t}_k := \left\{\frac{k\boldsymbol{z}}{N}\right\} = \frac{k\boldsymbol{z} \bmod N}{N}, \quad k=1,\dots,N, \tag{2}$$

其中，$\boldsymbol{z}\in\mathbb{Z}^s$ 是生成向量，向量两侧的花括号表示对向量中的每个分量取小数部分。等价地，这些点可以从 0 编号到 $N-1$，其中 $\boldsymbol{t}_0\equiv\boldsymbol{t}_N$。格点规则的质量取决于其生成向量 $\boldsymbol{z}$ 的选择。我们可以将每个分量 z_j 限定在集合 $\mathbb{Z}_N:=\{z\in\mathbb{Z}:\ 1\le z\le N-1 \text{ 且 } \gcd(z,N)=1\}$ 中，这确保格点规则的每个一维投影恰好是等间距点 $\{k/N:\ k=0,1,\dots,N-1\}$。请注意，不同维度之间出现重复分量可能不理想，因为这会导致低维投影质量不佳。图 1（左）展示了二维中的一个 64 点格点规则（$s=2$）。

随机平移格点规则（或更一般地说，随机平移 QMC 方法）具有如下形式

$$Q(F,\boldsymbol{\Delta}) := \frac{1}{N}\sum_{k=1}^{N} F(\{\boldsymbol{t}_k+\boldsymbol{\Delta}\}),$$

其中 $\boldsymbol{\Delta}\in[0,1]^s$ 是由 $[0,1]$, 均匀生成、各分量相互独立的平移，花括号表示如上所述取小数部分。图 1（右）展示了将二维 64 点格点规则按 $\boldsymbol{\Delta}=(0.1,\ 0.3)$ 平移后的结果。随机平移可得到积分的无偏估计量，因为 $\mathbb{E}[Q(F,\ \cdot)]=I(F)$。对该

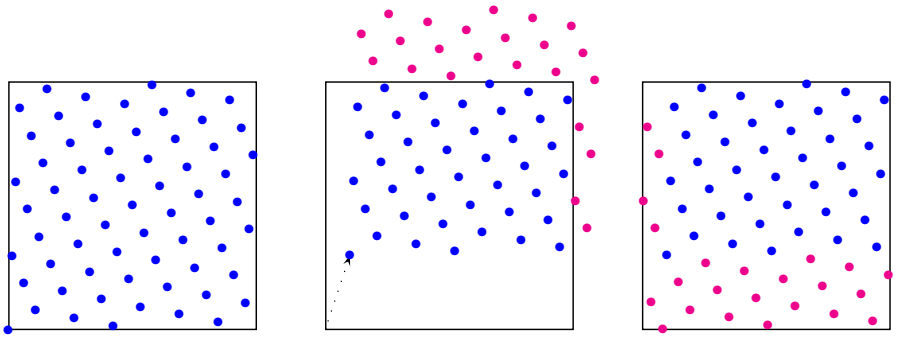


图 1：将（0.1,0.3）平移应用于二维中的 64 点格点规则。左：原始格点规则。中：将所有点平移（0.1,0.3）。右：将点重新包回单位立方体内。

相同的确定性规则进行多次（例如 16 次）独立随机平移，可提供实用的误差估计，参见例如 [21] 以及第 4 节中的数值实验。

格点也可用于函数逼近。我们将此类方法称为基于格点的算法。考虑定义在 $[0,1]^s$ 上、具有绝对收敛傅里叶级数的一周期实值 L_2 函数

$$F(\boldsymbol{y}) = \sum_{\boldsymbol{h}\in\mathbb{Z}^s} \hat{F}_{\boldsymbol{h}}\, e^{2\pi \mathrm{i}\boldsymbol{h}\cdot\boldsymbol{y}}, \quad \text{with} \quad \hat{F}_{\boldsymbol{h}} := \int_{[0,1]^s} F(\boldsymbol{y})\, e^{-2\pi \mathrm{i}\boldsymbol{h}\cdot\boldsymbol{y}}\, \mathrm{d}\boldsymbol{y},$$

其中 $\boldsymbol{h}\cdot\boldsymbol{y} = h_1 y_1 + \cdots + h_s y_s$ 表示通常的点积。不可能对无穷多个指标 $\boldsymbol{h}\in\mathbb{Z}^s$ 计算该求和。此外，傅里叶系数 $\hat{}\, F_{\boldsymbol{h}}$ 本身也是积分，需要加以近似。因此，一种基于格点的策略来近似 F，是先将傅里叶级数截断到一个精心选取的有限指标集 $\mathrm{A}_s \subset \mathbb{Z}^s$，再用具有 N 个点和生成向量 $\boldsymbol{z}\in\mathbb{Z}_N^s$ 的格点规则来近似 $\boldsymbol{h}\in\mathrm{A}_s$ 的傅里叶系数，从而得到截断三角级数（参见例如 [12, 13, 53, 59, 60] 和 [9, 43, 44, 80, 52]）

$$A^{\dagger}(F)(\boldsymbol{y}) := \sum_{\boldsymbol{h}\in\mathcal{A}_s} \hat{F}^a_{\boldsymbol{h}}\, e^{2\pi \mathrm{i}\boldsymbol{h}\cdot\boldsymbol{y}}, \quad \text{with} \quad \hat{F}^a_{\boldsymbol{h}} := \frac{1}{N}\sum_{k=1}^{N} F(\boldsymbol{t}_k)\, e^{-2\pi \mathrm{i}\boldsymbol{h}\cdot\boldsymbol{t}_k}. \tag{3}$$

得益于格点的结构，对所有 $\boldsymbol{h}\in\mathrm{A}_s$ 计算 $\hat{}\, F^a_{\boldsymbol{h}}$ 可借助一维快速傅里叶变换（FFT）在 $\mathrm{O}(N \log N)$ 次运算内完成。随后，计算$A^{\dagger}(F)$的代价取决于指标集 A_s 的结构与规模。

假设 F 属于一个核为 K 的再生核希尔伯特空间。那么，另一种基于格点的近似 F 的策略是使用核方法 （例如参见 [40, 42, 54, 92, 93]）

$$A^{*}(F)(\boldsymbol{y}) := \sum_{k=1}^{N} a_k\, K(\boldsymbol{t}_k, \boldsymbol{y}), \tag{4}$$

其中，系数 a_k 通过求解以下线性方程组获得

$$F(\boldsymbol{t}_{k'}) = \sum_{k=1}^{N} a_k\, K(\boldsymbol{t}_k, \boldsymbol{t}_{k'}) \quad \text{for all } k' = 1, \ldots, N, \tag{5}$$

这确保核近似在格点处插值 F。如果再生核是平移不变的，即 $K(\boldsymbol{y},\ \boldsymbol{y}') = K(\{\boldsymbol{y}-\boldsymbol{y}'\},\ \boldsymbol{0})$，例如采用三角基函数的傅里叶级数就是如此，那么结合格点的结构，线性方程组 (5) 涉及一个循环矩阵

因此可使用 FFT 在 $O(N \log N)$ 次运算内求出系数 a_k。此时，计算 $A^*(F)$ 的成本取决于核 K 的定义。

与积分问题中的格点规则一样，基于格点的逼近算法 $A^\dagger(F)$ 和 $A^*(F)$ 的质量取决于格点生成向量 $\boldsymbol{z}$ 的选择（对于 $A^\dagger(F)$，还取决于指标集 A_s 的选择）。在积分和逼近的各种情形下，我们知道如何逐分量构造生成向量，以获得良好的理论误差界：通过归纳，选择格点生成向量 $\boldsymbol{z}$ 的第 j 个分量 $z_j \in \mathbb{Z}_N$，使其最小化某个误差准则或满足所需性质，同时固定先前选定的分量 $z_1, \ldots\ldots, z_{j-1}$。参见上文引用的论文及其中的参考文献；其中大多数采用了 [75] 提出的“快速”构造方法，该方法利用 FFT 处理格点的结构。我们将在接下来的小节中回顾其中一些情形。

2.1 用于积分的格点规则

将被积函数 F 的底层函数空间统称为 W，我们将该函数空间中具有生成向量 $\boldsymbol{z}$ 的 N 点格点规则 (2)（或更一般地，准蒙特卡洛方法 (1)）的最坏情况误差定义为

$$e_N^{\mathrm{wor}}(\boldsymbol{z}) := \sup_{F \in \mathcal{W},\, \|F\|_{\mathcal{W}} \le 1} |I(F) - Q(F)|,$$

其中 $\|\cdot\|_{\mathcal{W}}$ 表示 W 的范数。于是我们得到一般的积分误差界

$$|I(F) - Q(F)| \le e_N^{\mathrm{wor}}(\boldsymbol{z})\, \|F\|_{\mathcal{W}} \qquad \text{for all } F \in \mathcal{W}. \tag{6}$$

这样的误差界将被积函数 F 的影响与生成向量 $\boldsymbol{z}$的影响分离开来。在经典准蒙特卡罗理论中，著名的 Koksma–Hlawka不等式具有相同的形式（6）。

按照 [24, 86], , 当代准蒙特卡罗研究采用带权函数空间。每个变量子集 $\boldsymbol{y}_{\mathfrak{u}}$都对应一个权重参数 $\gamma_{\mathfrak{u}} > 0$：$=(y_j)_{j\in\mathfrak{u}}$，其中 $\mathfrak{u} \subseteq \{1 : s\} :=\{1, 2, \ldots\ldots, s\}$；这些权重共同调节不同变量子集的相对重要性。$\gamma_{\mathfrak{u}}$ 的值较小，意味着函数对 $\boldsymbol{y}_{\mathfrak{u}}$的依赖较弱。这些权重 $\boldsymbol{\gamma} := (\gamma_{\mathfrak{u}})_{\mathfrak{u}\subseteq\{1:s\}}$不仅出现在范数 $\|\cdot\|_{\mathcal{W}}$ 的定义中，也出现在最坏情况误差$e_N^{\mathrm{wor}}(\boldsymbol{z})$的公式中。给定权重后，我们的策略是通过逐分量构造，选择格点生成向量 $\boldsymbol{z}$，尽可能地减小最坏情况误差 $e_N^{\mathrm{wor}}(\boldsymbol{z})$，正如本节前面简要讨论的那样。所得误差界可通过归纳法自然地证明。有时我们会考虑随机移位格点规则，此时误差界在随机平移下以均方根（r.m.s.）意义成立。

我们总结格点规则的三种理论设定，这些设定将在本综述的后文中用到。下文记 $\partial_{\mathfrak{u}}^{\alpha}$: $= \prod_{j\in\mathfrak{u}} \left(\frac{\partial}{\partial y_j}\right)^{\alpha}$，并且

$$\varrho_\alpha(\lambda) := \frac{2\zeta(2\alpha\lambda)}{(2\pi)^{2\alpha\lambda}}, \qquad \lambda > \tfrac{1}{\alpha}.$$

其中 $\zeta(x) := \sum_{h=1}^{\infty} h^{-x}$ 是黎曼ζ函数。

(a) [21, 定理 5.8] 对于在具有一阶光滑度的（非周期、“非锚定”）加权索伯列夫空间 $\mathrm{W}1,_{\boldsymbol{\gamma}}$ 中使用的随机平移格点规则，其范数为

$$\|F\|_{\mathcal{W}_{1,\boldsymbol{\gamma}}}^2 := \sum_{\mathfrak{u}\subseteq\{1:s\}} \frac{1}{\gamma_{\mathfrak{u}}} \int_{[0,1]^{|\mathfrak{u}|}} \left| \int_{[0,1]^{s-|\mathfrak{u}|}} \partial_{\mathfrak{u}}^{1} F(\boldsymbol{y})\, \mathrm{d}\boldsymbol{y}_{\{1:s\}\setminus\mathfrak{u}} \right|^2 \mathrm{d}\boldsymbol{y}_{\mathfrak{u}}, \tag{7}$$

可以构造一个良好的生成向量 $\boldsymbol{z}$，使得

$$\text{r.m.s. } e_N^{\mathrm{wor}}(\boldsymbol{z}) \le \left(\frac{2}{N} \sum_{\emptyset\ne\mathfrak{u}\subseteq\{1:s\}} \gamma_{\mathfrak{u}}^{\lambda} \left[\varrho_1(\lambda)\, 2^{\lambda}\right]^{|\mathfrak{u}|} \right)^{\frac{1}{2\lambda}} \quad \forall\, \lambda \in (\tfrac{1}{2}, 1]. \tag{8}$$

(b) [21, 定理 5.12] 对于加权 Korobov 空间 $\mathrm{W}_{\alpha,\,\boldsymbol{\gamma}}$ 中的格点规则（周期性、Hilbert）而言，若光滑度参数 $\alpha \geq 1$为整数，且范数为

$$\|F\|^2_{\mathcal{W}_{\alpha,\boldsymbol{\gamma}}} := \sum_{\mathfrak{u}\subseteq\{1:s\}} \frac{1}{\gamma_{\mathfrak{u}}} \int_{[0,1]^{|\mathfrak{u}|}} \left| \int_{[0,1]^{s-|\mathfrak{u}|}} \partial^{\alpha}_{\mathfrak{u}} F(\boldsymbol{y})\, \mathrm{d}\boldsymbol{y}_{\{1:s\}\setminus\mathfrak{u}} \right|^2 \mathrm{d}\boldsymbol{y}_{\mathfrak{u}}, \tag{9}$$

可以构造一个良好的生成向量 $\boldsymbol{z}$，使得

$$e^{\mathrm{wor}}_N(\boldsymbol{z}) \leq \left(\frac{2}{N} \sum_{\emptyset\neq\mathfrak{u}\subseteq\{1:s\}} \gamma^{\lambda}_{\mathfrak{u}} \left[\varrho_{\alpha}(\lambda)\right]^{|\mathfrak{u}|} \right)^{\frac{1}{2\lambda}} \quad \forall\, \lambda \in (\tfrac{1}{2\alpha}, 1]. \tag{10}$$

(c) [41, 引理 3.1]对于（周期、非希尔伯特）加权 Korobov 空间$\mathrm{K}_{\alpha,\boldsymbol{\gamma}}$ 中的格点规则，其整数光滑度参数为 $\alpha \geq 2$，且具有范数界

$$\|F\|_{\mathcal{K}_{\alpha,\boldsymbol{\gamma}}} \leq \max_{\mathfrak{u}\subseteq\{1:s\}} \frac{1}{\gamma_{\mathfrak{u}}} \int_{[0,1]^{|\mathfrak{u}|}} \left| \int_{[0,1]^{s-|\mathfrak{u}|}} \partial^{\alpha}_{\mathfrak{u}} F(\boldsymbol{y})\, \mathrm{d}\boldsymbol{y}_{\{1:s\}\setminus\mathfrak{u}} \right| \mathrm{d}\boldsymbol{y}_{\mathfrak{u}}, \tag{11}$$

可以构造一个良好的生成向量 $\boldsymbol{z}$，使得

$$e^{\mathrm{wor}}_N(\boldsymbol{z}) \leq \left(\frac{2}{N} \sum_{\emptyset\neq\mathfrak{u}\subseteq\{1:s\}} \gamma^{\lambda}_{\mathfrak{u}} \left[\varrho_{\alpha}(\lambda/2)\right]^{|\mathfrak{u}|} \right)^{\frac{1}{\lambda}} \quad \forall\, \lambda \in (\tfrac{1}{\alpha}, 1]. \tag{12}$$

上述三项结果均是在 N 为素数的幂的情形下建立的，并且对任意合数 N也有类似结果。此外，可以构造“嵌入式”格点规则 [11]，使其适用于一系列 N 取值（例如从 2^{10} 到 2^{20}），但代价是在误差界中引入一个较小的额外常数因子。

我们有意以这种方式呈现结果，以确保当 $\alpha = 1$时，非周期情形 (a) 的范数 (7) 与周期情形 (b) 的范数 (9) 相一致。希尔伯特 Korobov 范数通常根据傅里叶系数的衰减来定义，其中实参数为 $\alpha > 1/2$。仅当 α 为整数时，该范数才能如 (9) 那样用混合导数表示。在某些情形中，参数 α 会按 1/2 进行缩放，并且因子 $(2\pi)^{2\alpha|\mathfrak{u}|}$ 常常被吸收进权重 $\gamma_{\mathfrak{u}}$中。类似地，非希尔伯特 Korobov 范数通常通过带实参数$\alpha > 1$的傅里叶级数衰减来定义，并且仅当 α 取整数值时，它才能满足形如 (11) 的、涉及混合导数的界。在经典格点规则理论 [84], 中，相应的最坏情况误差常被称为 P_{α}。这里，我们使用记号 $\mathrm{K}_{\alpha,\boldsymbol{\gamma}}$，以将该空间与范数 (7) 和 (9) 所关联的空间区分开来，从而避免可能的混淆。

设定 (a) 可达到接近 $\mathrm{O}(N^{-1})$ 的收敛速率。相比之下，设定 (b) 和 (c) 均可达到接近 $\mathrm{O}(N^{-\alpha})$ 的收敛速率，但为确保相关常数不依赖于输入维数 s，对权重 $\gamma_{\mathfrak{u}}$ 所施加的条件可能大相径庭。在定理 3 中，我们将重点介绍非希尔伯特情形的一项独特优势。

格点规则还有许多其他理论，包括帐篷变换格规则（例如参见 [14, 22, 39]）、无界区域 $\mathbb{R}^s$ 上的格点规则（例如参见 [73, 76]）、具有随机点数的格点规则（例如参见 [18, 51, 56]），以及基于 均值中位数 的格点规则（例如参见 [28]）。

2.2 基于格点的函数逼近算法

现在我们转向函数逼近问题。再次将目标函数 F 所属的函数空间统称为 W，我们在该函数空间中定义基于格的算法 $A(F)$ 的最坏情况 L_2 逼近误差 为

$$e^{\mathrm{wor-app}}_N(A, \boldsymbol{z}) := \sup_{F\in\mathcal{W},\, \|F\|_{\mathcal{W}}\leq 1} \|F - A(F)\|_{L_2},$$

其中， $A(F)$ 可以是例如式 (3) 中定义的截断三角级数 $A^\dagger(F)$，也可以是式 (4) 中定义的核方法 $A^*(F)$。于是，我们得到与式 (6) 形式相同的通用逼近误差界

$$\|F - A(F)\|_{L_2} \le e_N^{\text{wor-app}}(A, \boldsymbol{z})\, \|F\|_{\mathcal{W}} \qquad \text{for all } F \in \mathcal{W}.$$

众所周知，在使用相同点集的所有算法中，核方法在最坏情形逼近误差意义下是最优的。因此

$$e_N^{\text{wor-app}}(A^*, \boldsymbol{z}) \le e_N^{\text{wor-app}}(A^\dagger, \boldsymbol{z}) \qquad \text{for all } \boldsymbol{z} \in \mathbb{Z}_n^s.$$

我们知道如何为截断三角级数 $A^\dagger(F)$ 构造优良的生成向量：可以直接通过最小化最坏情形逼近误差 $e_N^{\text{wor-app}}(A^\dagger, \boldsymbol{z})$ 的上界来实现（例如参见 [12, 13, 53, 59, 60]），也可以使用重构格（例如参见 [9, 43, 44, 80, 52]）。这两种方法得到的生成向量 $\boldsymbol{z}$ 随后都可用于核方法 $A^*(F)$。文献 [54] 对核方法中一种基于幂函数的替代误差准则进行了实验。在上述 (b) 函数空间设定下，逼近误差的收敛速率接近 $\mathrm{O}(N^{-\alpha}/2)$。文献 [9] 已证明，使用格点所能达到的最佳速率就是这一速率；相比之下，其他非基于格的算法所能达到的最佳速率接近 $\mathrm{O}(N^{-\alpha})$。尽管如此，基于格点的算法仍具有竞争力，因为它们能够实现简单且高效的快速傅里叶变换实现。文献 [7] 已证明，子采样格可以提高收敛速率。对于足够光滑的函数，核方法能够实现“速率翻倍”，参见 [85]。

3 深度神经网络作为函数逼近算法

现在，我们转而讨论作为非线性函数逼近器的深度神经网络（DNN）（例如参见 [29, 61]）。我们将第 2 节中的基于格的逼近方法推广到在格点采样的数据上训练的深度神经网络。某些应用会提供固定的训练数据，而另一些则允许从业者自行选择采样点，例如，为计算成本高昂的函数构建深度神经网络代理模型。我们总结近期论文 [47] 中关于深度神经网络正则性的结果，并给出新的推广。

我们不再局限于像此前讨论的那样处理实值函数 $F(\boldsymbol{y})$，而是允许输出为向量值；也就是说，我们关注一般形式的问题：

$$\text{Given data } \boldsymbol{y} \in Y \subset \mathbb{R}^s, \text{ compute observable } G(\boldsymbol{y}) \in Z = \mathbb{R}^{N_{\text{obs}}}.$$

例如，若 $\boldsymbol{y}$ 是参数化偏微分方程的 s 个输入参数构成的向量，那么 $G(\boldsymbol{y})$ 就是由偏微分方程解得到的可观测量向量，它是关于 $\boldsymbol{y}$ 的一个较为光滑的函数。可观测量可以是偏微分方程解在定义域某个子集上的平均值，此时 $N_{\text{obs}} = 1$；也可以是偏微分方程解在有限元网格各点处的取值构成的向量，此时 N_{obs} 可以是有限元网格内部点的数量。

3.1 非周期和周期深度神经网络

对于输入 $\boldsymbol{y} \in Y := [0,1]^s$，我们考虑深度神经网络（DNN）的两种形式：标准前馈深度神经网络，以及一种专为周期目标函数设计的专用架构，近期在 [47]中提出：

(a) 非周期深度神经网络：

$$G_\theta^{[L]}(\boldsymbol{y}) := W_L\, \sigma(\cdots W_2\, \sigma(W_1\, \sigma(W_0\, \boldsymbol{y} + \boldsymbol{v}_0) + \boldsymbol{v}_1) + \boldsymbol{v}_2 \cdots) + \boldsymbol{v}_L. \tag{13}$$

(b) 周期深度神经网络：

$$G_\theta^{[L]}(\boldsymbol{y}) := W_L\, \sigma(\cdots W_2\, \sigma(W_1\, \sigma(W_0\, \sin(2\pi \boldsymbol{y}) + \boldsymbol{v}_0) + \boldsymbol{v}_1) + \boldsymbol{v}_2 \cdots) + \boldsymbol{v}_L. \tag{14}$$

在此背景下， $W_\ell \in \mathbb{R}^{d_{\ell+1}\times d_\ell}$ 是权重矩阵， $\boldsymbol{v}_\ell \in \mathbb{R}^{d_{\ell+1}\times 1}$ 是偏置向量， L 是神经网络的深度。由 $l = 0$ 索引的输入层从输入向量 $\boldsymbol{y}$ 继承其维度 $d_0 = s$ ，而输出层 $l = L+1$ 的维度 $d_{L+1} = N_{\text{obs}}$ 是可观测量的数量。对于 $1 \le l \le N$ ，一层称为隐藏层，其宽度为 d_ℓ 。网络参数的全体 $\theta := (W_\ell, \boldsymbol{v}_\ell)_{\ell=0}^{L} \in \Theta$ 描述了一个全连接神经网络（也称为多层感知机）。所有允许的网络参数构成的集合记为 Θ 。 $N_{\text{DNN}} := \sum_{\ell=0}^{L}(d_{\ell+1}\times d_\ell) + \sum_{\ell=0}^{L} d_{\ell+1}$ 参数需要进行训练。深度 L 、宽度 d_ℓ 以及训练阶段所用的其他参数（见下文）的总体称为超参数。

标量非线性激活函数 σ 逐元素作用于由仿射变换得到的向量。常见的标量激活函数将在下文的引理1中介绍并讨论。通过对输入层逐元素施加正弦函数来扩展（13），即得到（14）。该正弦函数可视为周期激活函数 [83]。事实上，非线性函数（14）关于输入参数 $\boldsymbol{y}$ 是以1为周期的，正如第 2 节中理论设定（b）和（c）所示。特定形式（14）旨在处理目标函数 $G(\boldsymbol{y})$ ，该目标函数本身是 $\sin(2\pi \boldsymbol{y})$ 的函数，例如来自 [37, 40, 41, 42] 的感兴趣量。

训练深度神经网络（深度神经网络）需要确定合适的网络参数 θ ，使得 $G_\theta^{[L]}(\boldsymbol{y})$ 能够准确逼近 $G(\boldsymbol{y})$ ，尤其是在未包含于训练集中的点上。训练集由点 $\boldsymbol{y}_1, \ldots\ldots, \boldsymbol{y}_N \in Y$ 及其相应的函数求值 $G(\boldsymbol{y}_1), \ldots\ldots, G(\boldsymbol{y}_N)$ 组成，这些函数求值充当可观测量。在本工作中，我们将依赖合成生成数据（合成生成数据），具体采用格点进行训练。

3.2 误差

“训练”通过最小化适当的“损失函数” $\mathrm{J}(\theta)$ 来完成。遵循 [47] （而 [47] 又遵循 [66, 67, 69]），我们考虑

$$\theta^* := \operatorname*{argmin}_{\theta\in\Theta}\Big(\mathcal{J}(\theta) + \lambda\mathcal{R}(\theta)\Big), \quad \mathcal{J}(\theta) := \frac{1}{N}\sum_{k=1}^{N}\big\|G(\boldsymbol{y}_k) - G_\theta^{[L]}(\boldsymbol{y}_k)\big\|_2^2, \tag{15}$$

其中，我们对可观测量中的 N_{obs} 分量取向量二范数， $\mathrm{R}(\theta)$ 是一个正则化项，我们将根据理论对其进行量身定制。深度神经网络的质量以泛化误差（或 L_2 逼近误差）来表示

$$\mathcal{E}_G := \left(\int_Y \big\|G(\boldsymbol{y}) - G_\theta^{[L]}(\boldsymbol{y})\big\|_2^2 \,\mathrm{d}\boldsymbol{y}\right)^{1/2} \le \mathcal{E}_T + |\mathcal{E}_G - \mathcal{E}_T|, \tag{16}$$

其中， $\mathrm{E}_T := \sqrt{\mathrm{J}(\theta)}$ 是可计算的训练误差，而 $|\mathrm{E}_G - \mathrm{E}_T|$ 则是所谓的泛化差距。 E_G 和 E_T 中的平方根使得直接分析泛化差距变得困难。因此，我们采用较宽松的估计 $|\mathrm{E}_G - \mathrm{E}_T| \le \sqrt{|\mathrm{E}_G^2 - \mathrm{E}_T^2|}$ ，其中

$$\begin{aligned}
|\mathcal{E}_G^2 - \mathcal{E}_T^2| &= \left|\int_Y \big\|G(\boldsymbol{y}) - G_\theta^{[L]}(\boldsymbol{y})\big\|_2^2 \,\mathrm{d}\boldsymbol{y} - \frac{1}{N}\sum_{k=1}^{N}\big\|G(\boldsymbol{y}_k) - G_\theta^{[L]}(\boldsymbol{y}_k)\big\|_2^2\right| \\
&\le \sum_{p=1}^{N_{\text{obs}}}\left|\int_Y \big(G(\boldsymbol{y})_p - G_\theta^{[L]}(\boldsymbol{y})_p\big)^2 \,\mathrm{d}\boldsymbol{y} - \frac{1}{N}\sum_{k=1}^{N}\big(G(\boldsymbol{y}_k)_p - G_\theta^{[L]}(\boldsymbol{y}_k)_p\big)^2\right| \\
&\le \sum_{p=1}^{N_{\text{obs}}} e_N^{\text{wor}}(\boldsymbol{z})\,\big\|\big(G(\cdot)_p - G_\theta^{[L]}(\cdot)_p\big)^2\big\|_{\mathcal{W}},
\end{aligned} \tag{17}$$

其中，对于每个分量 p ，我们都有一个形式如 (6) 的一般误差界，即函数空间 W 中的格点规则（具有 N 个点和生成向量 $\boldsymbol{z}$ ）的最坏情况积分误差 $e_N^{\text{wor}}(\boldsymbol{z})$ ，乘以标量值被积函数 $F(\boldsymbol{y}) := (G(\boldsymbol{y})_p - G_\theta^{[L]}(\boldsymbol{y})_p)^2$ 在 W 中的范数。

现在，我们可以应用第2.1节中的理论设定 (a)–(c)，但在每种设定下，都需要证明被积函数 $(G(\cdot)_p - G_\theta^{[L]}(\cdot)_p)^2$ 确实属于相应的加权函数空间，且其范数有良好的界。由于这三种设定中的范数均以混合导数定义，因此我们需要关于深度神经网络相对于输入参数 $\boldsymbol{y} \in Y$的正则性界。

3.3 正则性

沿用 [47],，我们假设存在正序列 $(\beta_j)_{j\geq 1}$、$(R_\ell)_{\ell\geq 1}$、$(A_n)_{n\geq 1}$，使得以下三个关键假设成立：

1. 矩阵 W_0 的列具有有界的向量 ∞-范数

$$\|W_{0,:,j}\|_\infty := \max_{1\leq p\leq d_1} |W_{0,p,j}| \;\leq\; \beta_j \quad \text{for all} \quad j = 1,\ldots,s. \tag{18}$$

2. 对于$l \geq 1$，矩阵 W_ℓ 具有有界的矩阵 ∞-范数

$$\|W_\ell\|_\infty := \max_{1\leq p\leq d_{\ell+1}} \sum_{q=1}^{d_\ell} |W_{\ell,p,q}| \;\leq\; R_\ell \quad \text{for all} \quad \ell \geq 1. \tag{19}$$

3. 激活函数 $\sigma\colon \mathbb{R}\to\mathbb{R}$是光滑的，且其导数满足

$$\|\sigma^{(n)}\|_\infty := \sup_{x\in\mathbb{R}} |\sigma^{(n)}(x)| \;\leq\; A_n \quad \text{for all} \quad n \geq 1. \tag{20}$$

假设 (18) 和 (19) 类似于 [67, 命题 3.2 和 3.8] 中的假设。假设 (20) 允许采用相当一般的光滑激活函数。

具体而言，如同 [47] 中那样，我们将针对其导数在(20)的意义下由如下常见形式界定的激活函数来说明我们的结果

$$A_n = \xi\,\tau^n\,n! \qquad \text{for some } \xi > 0 \text{ and } \tau > 0. \tag{21}$$

下面的引理给出了一些常用激活函数的导数界。

引理1 对于 $c > 0$，下列广义激活函数满足(20)，其中 A_n 具有(21)的形式：

$$\begin{aligned}
\mathrm{sigmoid}_c(x) &:= \frac{1}{1+e^{-cx}}, && A_n = c^n\,n! && \textit{(i.e., } \xi = 1,\ \tau = c\textit{)};\\
\tanh_c(x) &:= \frac{e^{cx}-e^{-cx}}{e^{cx}+e^{-cx}}, && A_n = (2c)^n\,n! && \textit{(i.e., } \xi = 1,\ \tau = 2c\textit{)};\\
\mathrm{swish}_c(x) &:= \frac{x}{1+e^{-cx}}, && A_n = \tfrac{1.1}{c}\,c^n\,n! && \textit{(i.e., } \xi = \tfrac{1.1}{c},\ \tau = c\textit{)}.
\end{aligned}$$

证明。 标准定义下取 $c = 1$ 时的结果已在 [47]中得到证明。现在我们将这些结果推广到一般的 $c > 0$。我们有Sigmoid函数$_c(x)$ =Sigmoid函数$_1(cx)$，因此

$$\left|\mathrm{sigmoid}_c^{(n)}(x)\right| = c^n\left|\mathrm{sigmoid}_1^{(n)}(cx)\right| \leq c^n\,n!\,.$$

类似地，我们有双曲正切函数$_c(x)$ =双曲正切函数$_1(cx)$，因此

$$\left|\tanh_c^{(n)}(x)\right| = c^n\left|\tanh_1^{(n)}(cx)\right| \leq c^n\times 2^n\,n! = (2c)^n\,n!\,.$$

广义Swish函数与标准Swish函数的关系方式有所不同，即Swish函数$_c(x)$ =Swish函数$_1(cx)/c$，在这种情况下

$$\left|\mathrm{swish}_c^{(n)}(x)\right| = c^{n-1}\left|\mathrm{swish}_1^{(n)}(cx)\right| \leq c^{n-1}\times 1.1\,n! = \tfrac{1.1}{c}\,c^n\,n!\,.$$

证明至此完成。 □

注意，$\mathrm{ReLU}(x) := \max\{x,0\}$ 并不光滑，因此条件 (20) 不成立。然而，我们有

$$\mathrm{swish}_c(x) \to \mathrm{ReLU}(x) \quad \text{as} \quad c \to \infty.$$

显然，Swish函数$_c(0) = 0 =$ReLU(0)；对于 $x > 0$，当 $c \to \infty$时，Swish函数$_c(x) \to x =$ReLU(x)；对于 $x < 0$，当 $c \to \infty$时，Swish函数$_c(x) \to 0 =$ReLU(x)。当 $x \to +\infty$时，ReLU(x)与Swish函数$_c(x)$均无界。

下一引理表明，引理1中导数界的阶乘增长无法避免。这是因为 [47]中针对情形 $c = 1$的界均由标准S形函数的界推导而来，下文将给出该函数的一个下界。

引理2 对于 $\sigma(x) =$ Sigmoid函数$_1(x) = 1/(1+e^{-x})$，对所有 $n \geq 1$，我们有

$$\|\sigma^{(n)}\|_\infty \leq \frac{n^n}{(n+1)^{n+1}}\, n! < (n-1)! < n!,$$

并且对于所有奇数 $n \geq 1$，我们有下界

$$\|\sigma^{(n)}\|_\infty \geq |\sigma^{(n)}(\tfrac{1}{2})| = \frac{2(2^{n+1}-1)\,\zeta(n+1)}{(2\pi)^{n+1}}\, n! > \frac{1}{\pi^n}\, n!.$$

证明。 上界已在 [47]中利用恒等式 [68, 公式(15)]得到证明

$$\sigma^{(n)}(x) = \sum_{k=1}^{n} (-1)^{k-1} E(n,k-1)\,[\sigma(x)]^k\,[1-\sigma(x)]^{n+1-k}, \quad n \geq 1,$$

其中 $E(n,k)$ 是欧拉数。为得到一个下界，我们考虑 $x = 1/2$,

$$\|\sigma^{(n)}\|_\infty \geq |\sigma^{(n)}(\tfrac{1}{2})| = \frac{1}{2^{n+1}} \left| \sum_{k=1}^{n} (-1)^{k-1} E(n,k-1) \right| = \frac{1}{2^{n+1}} \frac{2^{n+1}(2^{n+1}-1)\,|B_{n+1}|}{n+1},$$

这里我们用到了这样一个性质：欧拉数的交错和与伯努利数之间的关系由 [77, 公式 (26.14.11)] 给出。注意，当 n 为偶数时，$B_{n+1} = 0$。因此，对于 $m \geq 1$，我们考虑奇数 $n = 2m-1$，并对 $|B_{n+1}| = |B_{2m}|$ 使用 [77,公式 (25.6.2)]，得到

$$|\sigma^{(n)}(\tfrac{1}{2})| = \frac{2^{n+1}-1}{n+1}\, \frac{2\zeta(2m)\,(2m)!}{(2\pi)^{2m}} = \frac{2(2^{n+1}-1)\,\zeta(n+1)}{(2\pi)^{n+1}}\, n! > \frac{1}{3\pi^{n-1}}\, n! > \frac{1}{\pi^n}\, n!,$$

其中黎曼ζ函数满足 $\zeta(n+1) \geq \zeta(2) = \pi^2/6$。 □

为了陈述深度神经网络的正则性结果，我们需要引入更多记号。令$\mathbb{N}_0$:={0,1,2,…} 为非负整数集。对于多重指标 $\boldsymbol{\nu} = (\nu_j)_{j\geq 1} \in \mathbb{N}_0^\infty$，我们将其阶定义为 $|\boldsymbol{\nu}| := \sum_{j\geq 1} \nu_j$，并考虑指标集 I :={$\boldsymbol{\nu} \in \mathbb{N}_0^\infty$: $|\boldsymbol{\nu}| < \infty$}。当对所有 $j \geq 1$ 都有 $m_j \leq \nu_j$ 时，我们记作 $\boldsymbol{m} \leq \boldsymbol{\nu}$。我们将关于参数变量 $\boldsymbol{y}$ 的混合偏导算子定义为 $\partial^{\boldsymbol{\nu}} := \prod_{j\geq 1} (\frac{\partial}{\partial y_j})^{\nu_j}$ 和 $\partial^{\boldsymbol{\nu}_\mathfrak{u}} := \prod_{j\in\mathfrak{u}} (\frac{\partial}{\partial y_j})^{\nu_j}$。我们还需要第二类斯特林数，它们对整数 n 和 k 定义为 S $(n,0) := \delta_{n,0}$（即，当 $n = 0$ 时其值为 1，当$n \geq 1$时其值为 0），$\mathrm{S}(n,k) := 0$（当 $k > n$ 时），否则

$$\mathcal{S}(n,k) := \frac{1}{k!} \sum_{i=0}^{k} (-1)^{k-i} \binom{k}{i} i^n, \qquad n \geq k.$$

定理 1（[47,定理 2。2]） 设序列 $(\beta_j)_{j\geq 1}$、$(R_\ell)_{\ell\geq 1}$、$(A_n)_{n\geq 1}$分别按照 (18)、(19)、(20) 定义，其中 A_n 由参数 ξ、$\tau > 0$按照 (21) 给出。对于深度 $L \geq 1$、任意分量 $1 \leq p \leq N_{\mathrm{obs}}$以及任意多重指标 $\boldsymbol{\nu} \in \mathrm{I}$（包括 $\boldsymbol{\nu} = \mathbf{0}$），有以下正则性界：

（a）由 （13）定义的非周期深度神经网络满足

$$|\partial^{\nu} G_{\theta}^{[L]}(\boldsymbol{y})_p| \le C_L\ |\boldsymbol{\nu}|! \prod_{j=1}^{s} (S_L \beta_j)^{\nu_j}. \tag{22}$$

(b) 在(14)中定义的周期深度神经网络满足

$$|\partial^{\nu} G_{\theta}^{[L]}(\boldsymbol{y})_p| \le C_L\,(2\pi)^{|\nu|} \sum_{\boldsymbol{m} \le \boldsymbol{\nu}} |\boldsymbol{m}|! \prod_{j=1}^{s} \Big((S_L \beta_j)^{m_j}\, \mathcal{S}(\nu_j, m_j) \Big). \tag{23}$$

在这两种情形下，我们定义

$$C_L := \max\left\{ \|G_{\theta}^{[L]}\|_{\infty},\ \frac{P_L}{S_L} \right\}, \quad S_L := \tau \sum_{\ell=0}^{L-1} P_{\ell}, \quad P_{\ell} := \prod_{t=1}^{\ell} (\xi\, \tau\, R_t). \tag{24}$$

我们可以限制深度神经网络的网络参数，使其与目标函数的正则性特征相匹配，从而得到(17)中范数的理论界。形如(29)的目标函数正则性界已出现在偏微分方程应用中，例如 [26, 31, 33, 36, 55, 57], ，而形如(31)的界已出现在例如 [20, 37, 40, 41, 42]中。在这些偏微分方程应用中，序列 b_j刻画了随着 j增大、对应于第 j个基函数的随机场波动幅度的衰减。相应地，该序列 b_j反映了变量y_j在我们的加权函数空间中重要性的衰减。

我们强调，定理 1 中关于深度神经网络的界是完全一般的。若目标函数并非来自上述偏微分方程应用，则应首先得到该目标函数的正则性界，然后设计一种可能不同的策略来限制网络参数，使其与目标函数的正则性特征相匹配。

定理 2（[47,定理 3。2]） 设序列 $(\beta_j)_{j\ge1}$、$(R_\ell)_{\ell\ge1}$、$(A_n)_{n\ge1}$分别如(18)、(19)、(20)中所定义，其中 A_n 由参数 ξ、$\tau > 0$按照(21) 给出。对于深度 $L \ge 1$，令 C_L和 S_L如(24)中所定义。给定一个序列 $(b_j)_{j\ge1}$和一个常数 $C > 0$，我们

- 限制矩阵 W_1、……、W_{L-1} 的元素，使得对于某个常数 $\rho > 0$，

$$R_\ell \le \rho \quad \textit{for all } \ell = 1, \dots, L-1, \tag{25}$$

$$\Big(\textit{thus } S_L \le \tau L \textit{ if } \xi\,\tau\,\rho = 1, \textit{ and } S_L \le \tau \frac{(\xi\,\tau\,\rho)^L - 1}{\xi\,\tau\,\rho - 1} \textit{ otherwise} \Big), \tag{26}$$

- 限制矩阵 W_0 的元素，使得

$$\beta_j \le \frac{b_j}{S_L} \quad \textit{for all } j = 1, \dots, s, \tag{27}$$

- 限制矩阵 W_1、……、W_L和向量 $\boldsymbol{v}_L$，使得

$$C_L \le C. \tag{28}$$

(a) 假设非周期目标函数 $G(\boldsymbol{y})$满足以下正则性界：对于所有多重指标$\boldsymbol{\nu} \in \mathcal{I}$ 以及所有分量 $1 \le p \le N_{\text{obs}}$，

$$|\partial^{\nu} G(\boldsymbol{y})_p| \le C\,|\boldsymbol{\nu}|! \prod_{j=1}^{s} b_j^{\nu_j}. \tag{29}$$

那么，满足限制条件 (25)–(28) 的非周期深度神经网络(13) 也满足相同的正则性界(29)，并且对于范数(7)，对所有分量 $1 \le p \le N_{\text{obs}}$，

$$\left\| \left(G(\cdot)_p - G_\theta^{[L]}(\cdot)_p\right)^2 \right\|_{\mathcal{W}_{1,\boldsymbol{\gamma}}}^2 \le 16\, C^4 \sum_{\mathfrak{u} \subseteq \{1:s\}} \frac{1}{\gamma_{\mathfrak{u}}} \left((|\mathfrak{u}|+1)! \prod_{j \in \mathfrak{u}} b_j \right)^2. \tag{30}$$

(b) 假设周期目标函数 $G(\boldsymbol{y})$ 满足以下正则性界：对于所有多重指标 $\boldsymbol{\nu} \in \mathcal{I}$ 以及所有分量 $1 \le p \le N_{\text{obs}}$，

$$|\partial^{\boldsymbol{\nu}} G(\boldsymbol{y})_p| \le C\,(2\pi)^{|\boldsymbol{\nu}|} \sum_{\boldsymbol{m} \le \boldsymbol{\nu}} |\boldsymbol{m}|! \prod_{j=1}^{s} \left(b_j^{m_j}\, \mathcal{S}(\nu_j, m_j) \right). \tag{31}$$

那么，满足限制条件 (25)–(28) 的周期深度神经网络 (14) 满足相同的正则性界 (31)，并且对于范数 (9)，对所有分量 $1 \le p \le N_{\text{obs}}$，我们有

$$\begin{aligned} &\left\| \left(G(\cdot)_p - G_\theta^{[L]}(\cdot)_p\right)^2 \right\|_{\mathcal{W}_{\alpha,\boldsymbol{\gamma}}}^2 \\ &\le 16\, C^4 \sum_{\mathfrak{u} \subseteq \{1:s\}} \frac{(2\pi)^{2\alpha|\mathfrak{u}|}}{\gamma_{\mathfrak{u}}} \left(\sum_{\boldsymbol{m}_{\mathfrak{u}} \le \boldsymbol{\alpha}_{\mathfrak{u}}} (|\boldsymbol{m}_{\mathfrak{u}}|+1)! \prod_{j \in \mathfrak{u}} \left(b_j^{m_j}\, \mathcal{S}(\alpha, m_j) \right) \right)^2, \end{aligned} \tag{32}$$

其中，$\boldsymbol{\alpha}_{\mathfrak{u}}$ 表示一个多重指标，其 $|\mathfrak{u}|$ 个分量均等于 α。

(c) 延续(b)，对于范数 (11)，对所有分量 $1 \le p \le N_{\text{obs}}$，我们有 ，

$$\begin{aligned} &\left\| \left(G(\cdot)_p - G_\theta^{[L]}(\cdot)_p\right)^2 \right\|_{\mathcal{K}_{\alpha,\boldsymbol{\gamma}}} \\ &\le 4\, C^2 \max_{\mathfrak{u} \subseteq \{1:s\}} \frac{(2\pi)^{\alpha|\mathfrak{u}|}}{\gamma_{\mathfrak{u}}} \sum_{\boldsymbol{m}_{\mathfrak{u}} \le \boldsymbol{\alpha}_{\mathfrak{u}}} (|\boldsymbol{m}_{\mathfrak{u}}|+1)! \prod_{j \in \mathfrak{u}} \left(b_j^{m_j}\, \mathcal{S}(\alpha, m_j) \right). \end{aligned} \tag{33}$$

序列 b_j 出现在范数界 (30)、(32) 和 (33) 中，并直接决定了下文定理中权重 $\gamma_{\mathfrak{u}}$ 的选择。关键在于，尽管较大的权重 $\gamma_{\mathfrak{u}}$ 会减小这些范数界，但同时也会增大最坏情况误差界 (8)、(10) 和 (12)。由于泛化误差由这些因素的乘积所界定（见 (17)），我们对 $\gamma_{\mathfrak{u}}$ 进行优化，以平衡这一权衡。在这三个情形中，该框架都能构造出收敛速率与输入维数 s 无关的格生成向量。

虽然输入维数 s 始终固定，但为了使误差界中的常数独立于 s，我们考虑一个可和的无限序列 b_j。下文定义的“可和指数”决定了收敛速率。形式为 (34) 的权重称为 POD（“乘积与阶数相关”）权重，而形式为 (36) 和 (38) 的权重称为 SPOD（“光滑度驱动的乘积与阶数相关”）权重。

定理 3（[47,定理 3。3]） 在与定理 *2* 相同的条件下，对于相应的设置 (a)、(b)、(c)，进一步假设存在一个“可和指数” $p^* \in (0,\ 1)$，使得 $\sum_{j \ge 1} b_j^{p^*} < \infty$。

(a) 按照第 *2* 节的设置 (a)，为带权重的 *Sobolev* 空间中的随机移位格点规则构造一个生成向量 $\boldsymbol{z}$，其权重为

$$\gamma_{\mathfrak{u}} := \left((|\mathfrak{u}|+1)! \prod_{j \in \mathfrak{u}} \left(\sqrt{\frac{(2\pi)^{2\lambda}}{2\zeta(2\lambda)\, 2^{\lambda}}}\, b_j \right) \right)^{\frac{2}{1+\lambda}},\ \lambda := \begin{cases} \frac{1}{2-2\delta}, \delta \in (0, \frac{1}{2}), \textit{if } p^* \in (0, \frac{2}{3}], \\ \frac{p^*}{2-p^*} \qquad\qquad \textit{if } p^* \in (\frac{2}{3}, 1). \end{cases} \tag{34}$$

若使用这些格点训练非周期深度神经网络 (13)，且训练误差达到阈值 $\texttt{tol} \in (0,\ 1)$，则泛化误差满足

$$\mathcal{E}_G \le \texttt{tol} + O\left(N^{-r/2}\right), \qquad r := \min\left(1-\delta,\ \frac{1}{p^*} - \frac{1}{2}\right), \quad \delta \in (0, \tfrac{1}{2}). \tag{35}$$

(b) 按照第 *2* 节中的构造，在加权 *Hilbert Korobov* 空间中，为格点规则构造生成向量 z，其权重为

$$\gamma_{\mathfrak{u}} := (2\pi)^{2\alpha|\mathfrak{u}|}\sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}}\left((|\boldsymbol{m}_{\mathfrak{u}}|+1)!\prod_{j\in\mathfrak{u}}\frac{b_j^{m_j}\mathcal{S}(\alpha,m_j)}{\sqrt{2\zeta(2\alpha\lambda)}}\right)^{\frac{2}{1+\lambda}},\ \alpha := \left\lfloor\tfrac{1}{p^*}+\tfrac{1}{2}\right\rfloor,\ \lambda := \tfrac{p^*}{2-p^*}. \tag{36}$$

如果使用这些格点训练周期深度神经网络(14)，且训练误差达到阈值$\texttt{tol}\in(0,\ 1)$，则泛化误差满足

$$\mathcal{E}_G \le \texttt{tol} + O\big(N^{-r/2}\big),\qquad r := \frac{1}{p^*}-\frac{1}{2}. \tag{37}$$

(c) 按照第 *2* 节中的构造方式(c)，为加权非希尔伯特 *Korobov* 空间中的格点规则构造生成向量 z，其权重为

$$\gamma_{\mathfrak{u}} := (2\pi)^{\alpha|\mathfrak{u}|}\sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}}(|\boldsymbol{m}_{\mathfrak{u}}|+1)!\prod_{j\in\mathfrak{u}}\big(b_j^{m_j}\mathcal{S}(\alpha,m_j)\big),\quad \alpha := \left\lfloor\tfrac{1}{p^*}\right\rfloor+1. \tag{38}$$

如果使用这些格点训练周期深度神经网络(14)，且训练误差达到阈值$\texttt{tol}\in(0,\ 1)$，则泛化误差满足

$$\mathcal{E}_G \le \texttt{tol} + O\big(N^{-r/2}\big),\qquad r := \frac{1}{p^*}. \tag{39}$$

在这三种情况下，大O界中的隐含常数均与 s 无关。

如果训练后未达到条件(27)，即对于某个 j，$\beta_j \le b_j/S_L$ 不成立，其中 S_L 定义见(24)，而是存在常数 $\kappa \ge 1/s_L$，使得

$$\beta_j \le \kappa\, b_j \quad \textit{for all } j = 1,\dots,s, \tag{40}$$

那么，界(30)、(32)、(33)仍然成立，只需将每个 b_j 替换为 $\kappa S_L\, b_j$。尽管格训练点是基于原始 b_j，使用权重(34)、(36)、(38)构造的，但大O界(35)、(37)、(39)现在仍然成立，不过隐含常数会增大，且依赖于 κS_L。特别是，(35)和(37)对应的增大后常数仍有界，且与 s 无关；而(39)对应的增大后常数则随 s 指数增长。

这些结果通过将 (30) 与 (8)、(32) 与 (10)、以及 (33) 与 (12) 结合得到，然后根据 p^* 选择权重和其他参数（例如 α 和 λ），以确保每种情况下得到的常数都有限。权重和收敛速率的推导遵循了近期几篇研究中的类似思路，包括 [55, 57] 中的情形 (a)、[40]中的情形 (b) 和 [41] 中的情形 (c)。由于 [47], 省略了详细推导，为了清晰完整起见，我们在附录中给出完整证明。

注意，设置 (c) 在理论上的优势是：r 的值更大，因此收敛速度更快。

3.4 定制正则化

我们训练深度神经网络，以获得

$$\theta^* := \operatorname*{argmin}_{\theta\in\Theta}\left(\frac{1}{N}\sum_{k=1}^{N}\big\|G(\boldsymbol{y}_k)-G_\theta^{[L]}(\boldsymbol{y}_k)\big\|_2^2+\lambda\,\|\theta\|_2^2+\lambda_1\,\mathcal{R}_1(\theta)\right).$$

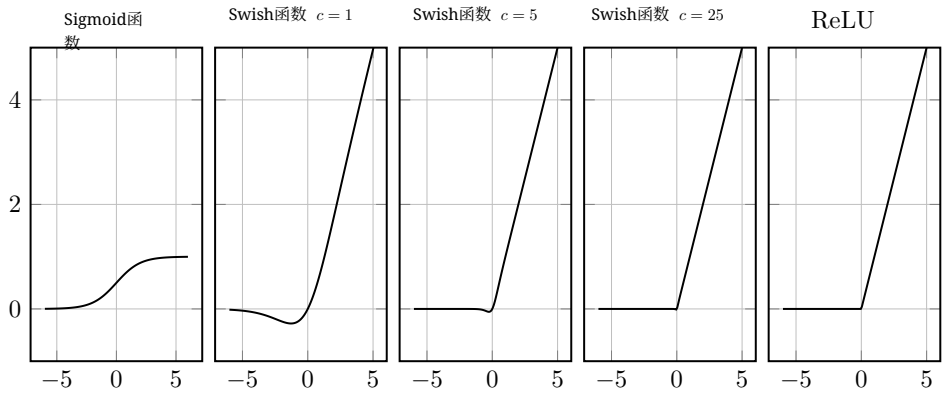


图 2：不同激活函数的图形。

当 $\lambda_1 = 0$ 时，我们采用标准的 l_2 正则化。当 $\lambda_1 > 0$ 时，我们采用 [47] 中的定制正则化，该正则化是基于定理 2 设计的：

$$\mathcal{R}_1(\theta) := \frac{1}{s}\sum_{j=1}^{s}\frac{1}{d_1}\sum_{p=1}^{d_1}\left(W_{0,p,j}^2\,\frac{L^2}{b_j^2}\right)^{m/2}, \tag{41}$$

其中 $m = 6$ 为正偶整数。特别地，选择 $\mathrm{R}_1(\theta)$ 是为了“鼓励”满足条件 (27)，即 $\|W_{0,j}\|_\infty \le \beta_j \le b_j/S_L$ 对于所有 $j = 1,\dots,s$，这等价于

$$\max_{1\le j\le s}\ \max_{1\le p\le d_1} |W_{0,p,j}|\,\frac{S_L}{b_j} \le 1. \tag{42}$$

为从 (42) 得到我们设计的正则化项 (41)，我们用 L 替换 S_L，以至少捕捉这样一个事实：S_L 通常会随着 L 的增加而增大；并且我们仅用 $m = 6$ 来估计 $\max_{1\le i\le n}|a_i| = \lim_{m\to\infty}\left(\frac{1}{n}\sum_{i=1}^{n}|a_i|^m\right)^{1/m}$，其中偶次幂使我们可以省去绝对值。$\mathrm{R}_1(\theta)$ 的光滑性使我们能够通过自动微分计算梯度 $\nabla_\theta \mathrm{R}_1(\theta)$。

[47] 中的数值实验表明，对于周期代数函数和 Sigmoid 激活函数，这种定制正则化能够有效促使序列 β_j 按照 b_j 的规律衰减；正如理论所预测的那样，与仅采用标准 l_2 正则化相比，这改善了泛化差距。

4 数值实验

在本研究中，我们将 [47] 中的实验扩展到 Sigmoid 函数以外的其他激活函数。具体而言，我们考察具有不同参数 $c \in (\{1,5,25\})$ 的 Swish$_c$ 函数，以及 ReLU 函数；后者并不光滑，因此不属于现有理论的适用范围。值得注意的是，当 $c \to \infty$ 时，$\mathrm{swish}_c(x)$ 收敛到 $\mathrm{ReLU}(x)$。与有界的 Sigmoid 函数不同，当 $x \to \infty$ 时，$\mathrm{swish}_c(x)$ 和 $\mathrm{ReLU}(x)$ 都无界。图 2 对这些函数进行了直观比较。由引理1和 (25)–(26) 可知，若对所有 $l = 1,\dots,L-1$ 均有 $R_\ell \le \rho$，则对于 $\mathrm{swish}_c(x)$，我们有

$$S_L \le c\,\frac{(1.1\rho)^L - 1}{1.1\rho - 1}.$$

因此，(22)–(23) 中出现的因子 S_L 会随着 c 的增加而线性增长，随后这种增长会因乘积而进一步放大。我们想了解这在实践中是否会影响性能。

与 [47], 一样，我们研究周期代数函数

$$G(\boldsymbol{y}) := \frac{1}{a(\boldsymbol{y})}, \quad a(\boldsymbol{y}) := 1 + \sum_{j=1}^{s} \sin(2\pi y_j)\, \psi_j, \quad \psi_j := \frac{\eta}{j^q} \text{ for } q > 1,$$

其满足正则性界 (31)，其中 $b_j = \psi_j / a_{\min}$ 和 $C = 1/a_{\min}$，且 $a_{\min} := 1 - \eta\, \zeta(q)$。在实验中，我们选择 $\eta = 0.5$ 和 $q = 2.5$。这对应于定理 3 中的 $p^* \approx 1/q = 1/2.5$，在设置 (a)、(b)、(c) 中分别给出预测的收敛速率 $r/2 \approx 0.5$、1、1.25。我们使用图 2 中的五种激活函数训练周期深度神经网络 (14)，并采用与 [47]相同的两组超参数：

1. $L = 3$, $N_{\text{obs}} = 1$, $d_\ell = 32$, $s = 50$ (3777 parameters);
2. $L = 12$, $N_{\text{obs}} = 1$, $d_\ell = 30$, $s = 50$ (11791 parameters).

尽管这些网络可以被认为规模较小，但近期研究已表明，“微型”神经网络在计算机图形学中取得了显著成功，见 [71, 72, 87]。

深度神经网络通过运行全批量 Adam（自适应矩估计）优化算法 [50], 进行训练，该算法是一种梯度下降法。我们的算法基于 Python 的开源深度学习框架 PyTorch [79]实现。随机 Glorot 初始化决定初始参数配置，训练时的学习率（步长）为 10^{-4}。当训练误差 $\mathrm{E}_T = \sqrt{\mathrm{J}(\theta)}$（见 (15)）达到阈值 tol $= 10^{-3}$，或完成最多 40000 个训练~~轮次~~（步）后，优化过程即告终止。

我们采用一个随机平移的现成嵌入格点集版本，该点集按照 [11], 构造，并将其用作训练点 $\{\boldsymbol{y}_k\}_{k\geq 1}$。网络使用 $N = 2^5, \ldots\ldots, 2^{12}$ 个点逐步进行训练。随后，我们将训练误差 E_T 与泛化误差 E_G的估计值（见 (16)）进行比较，该估计值由

$$\mathcal{E}_G \approx \widetilde{\mathcal{E}}_G := \left(\frac{1}{M} \sum_{i=1}^{M} \left(G(\boldsymbol{t}_i) - G_\theta^{[L]}(\boldsymbol{t}_i) \right)^2 \right)^{1/2},$$

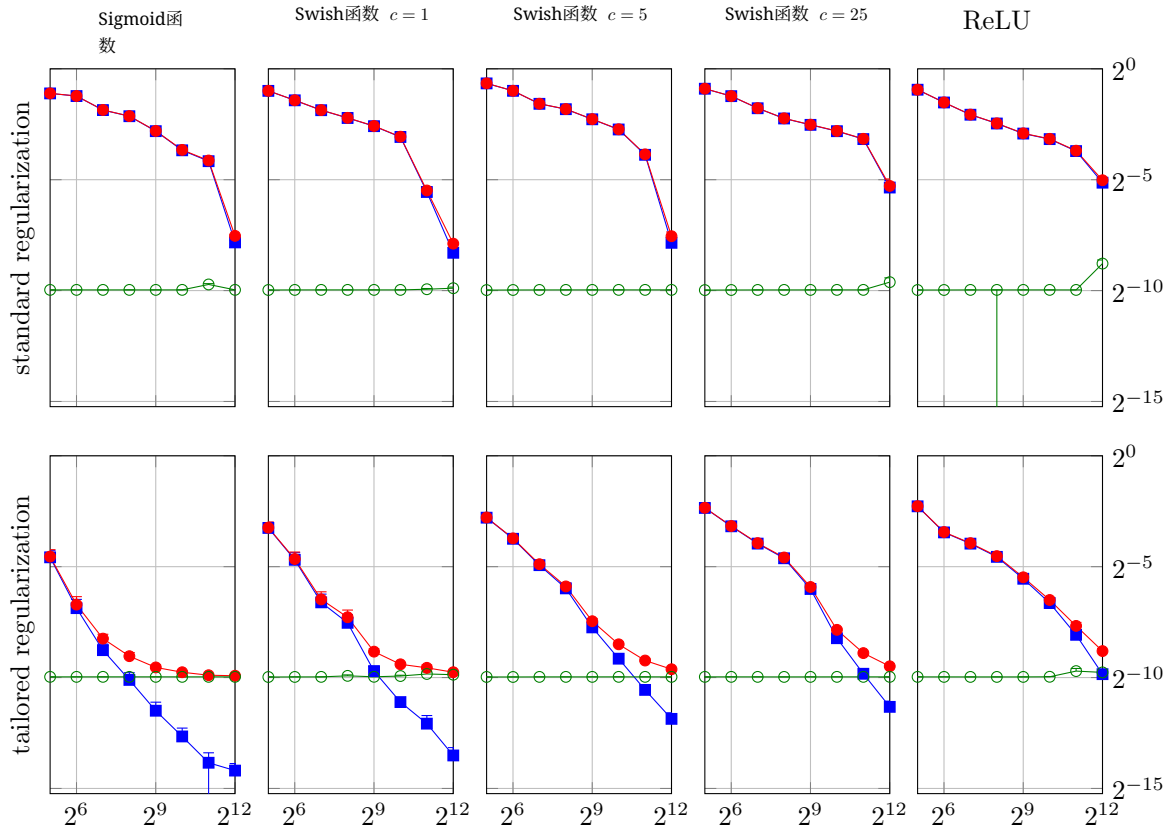
其中使用相同的嵌入格点规则，但采用不同且相互独立的随机平移，并使用数量大得多的点 $M = 2^{15} \gg N$。最后，我们通过 $|\mathrm{E}_G - \mathrm{E}_T|$ 计算泛化差距的估计值~。根据定理 3，我们知道

$$\mathcal{E}_G \leq \mathcal{E}_T + |\mathcal{E}_G - \mathcal{E}_T| \leq 10^{-3} + O\left(N^{-r/2}\right).$$

~应用定制正则化时，我们希望观察到始终较低的泛化误差 E_G~，以及差距 $|\mathrm{E}_G - \mathrm{E}_T|$ 更快的收敛速度。

在图3中，对于第一组超参数（$L = 3$），我们绘出了训练误差~$\mathcal{E}_T$ (○ 绿色圆圈）、估计的泛化误差 $\mathcal{E}_G$ (● 红点），以及泛化差距|~$\mathcal{E}_G - \mathcal{E}_T$|(■蓝色方块）的取值，适用于 $N = 2^5, \ldots, 2^{12}$。上述各个量均取网络 20 次不同随机 Glorot 初始化和格点规则随机平移的平均值。上方五幅图（标注为“标准正则化”，分别对应五种激活函数）使用正则化参数 $\lambda = 10^{-8}$，但未使用我们的定制正则化（因此 $\lambda_1 = 0$）。下方五幅图（标注为“定制正则化”）使用了 $\lambda = 10^{-8}$ 和我们的定制正则化$\lambda_1 = 10^{-8}$。我们通过反复试验选取 λ 和 λ_1 的取值，以确保性能始终出色。图4展示了第二组超参数（$L = 12$）对应的相同图形。

在图3和图4所示的全部五种激活函数中，我们观察到，与标准l_2 正则化相比，我们的定制正则化始终能提升性能。采用标准正则化时，所有估计的泛化误差 ~$\mathcal{E}_G$ (●) 均未达到我们期望的阈值tol $= 10^{-3}$，在 $N = 2^{12}$。



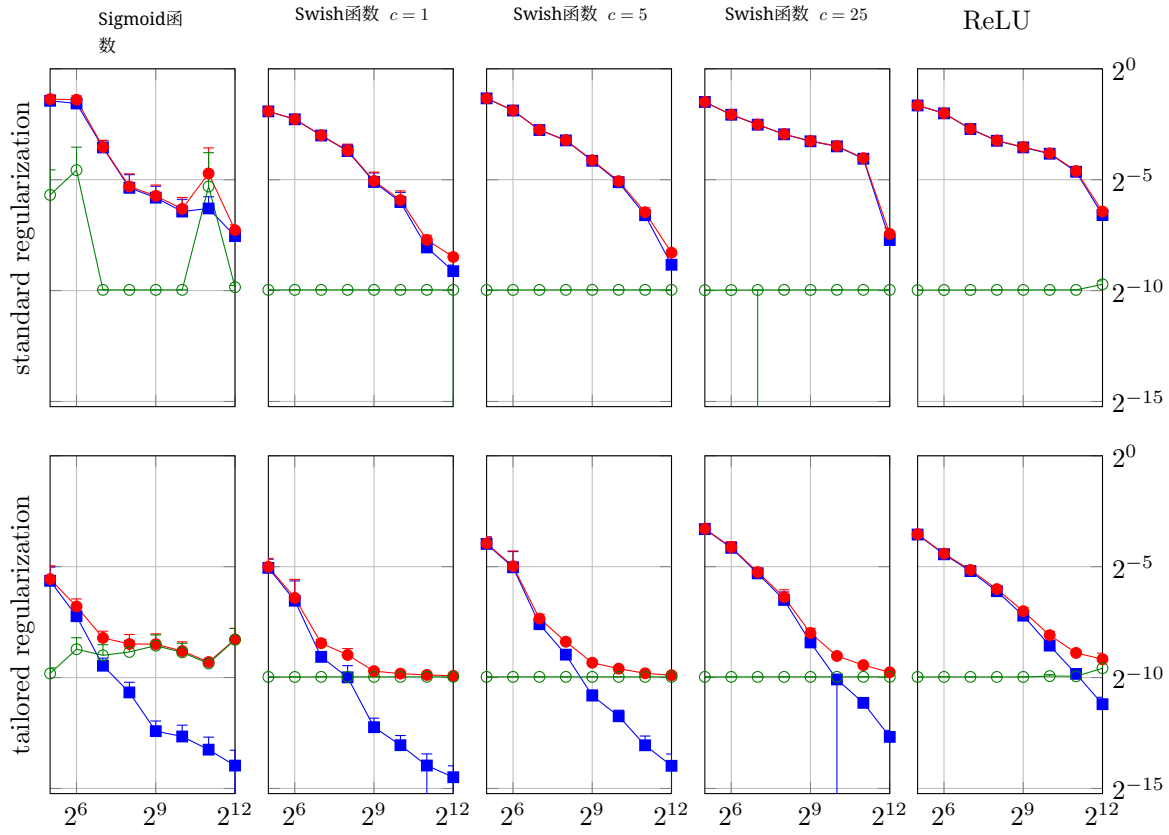
~ ~图3：数值 $\mathcal{E}_T$（○ 绿色圆圈）、$\mathcal{E}_G$（● 红点），以及 $|\mathcal{E}_G - \mathcal{E}_T|$（■ 蓝色方块）随着 N 的增加而变化，其中 $L = 3$，$N_{\text{obs}} = 1$，$d_\ell = 32$，$s = 50$。

相比之下，采用定制正则化时，显然，$c \in \{1, 5, 25\}$下 Swish函数的性能会随着 c 增大而下降，逐渐接近五种函数中表现最差的 ReLU。这一观察与我们的理论误差界相符，其中因子 S_L 随 c 增大；而 ReLU 缺乏光滑性，因此该理论不适用。

在图3（$L = 3$）中，Sigmoid激活函数脱颖而出，成为表现最佳者。此情形此前已在 [47, 图 1], 中~展示过；我们观察到，估计泛化间隙 $|\mathrm{E}_G - \mathrm{E}_T|$(■)以介于 $\mathrm{O}(N^{-1})$ 和 $\mathrm{O}(N^{-2})$ 之间的速率衰减，并在 $N = 2^9$时迅速降至 tol $= 10^{-3}$ 以下；从 $N = 2^{10}$ 开始，~E_G(●)则相当接近 tol $= 10^{-3}$。

图4（$L = 12$）中表现最佳的是标准Swish激活函数（$c = 1$），其间隙|~$\mathrm{E}_G - \mathrm{E}_T$|(■)以介于 $\mathrm{O}(N^{-1})$ 和 $\mathrm{O}(N^{-2})$ 之间的速率衰减，并且早在 $N = 2^9$时就已显著低于 tol $= 10^{-3}$，其中 ~E_G (●) 从 $N = 2^9$开始便与 tol $= 10^{-3}$ 相当接近。事实上，所有 Swish 和 ReLU 变体的表现都优于 Sigmoid函数。如 [47, 图 2]所示，在 Sigmoid函数的情形下，训练误差 $\mathrm{E}_T(\circ)$ 无法达到 tol $= 10^{-3}$。在训练过程中逐步降低学习率的策略或许能提升 Sigmoid函数在此情形下的表现。不过，值得注意的是，Swish函数在恒定学习率下始终表现出色。

在本综述中，我们不介绍标准蒙特卡洛（MC）点的结果。如 [47, 图 1和 2], 所示，MC 的表现与 QMC 惊人地相似。一种可能的解释是，定制正则化降低了（$G - G_\theta^{[L]}$）^{2}的方差。此外，如 [47, 图 3和 4],所示，对于所有激活函数，采用定制正则化的序列（β_j）都近似沿着序列（b_j/L）衰减；我们不在此重复这些图。



~ ~图4：的值$\mathcal{E}_T$(○（绿色圆圈）、$\mathcal{E}_G$(●（红点），以及$|\mathcal{E}_G - \mathcal{E}_T|$(■（蓝色方块），随着的增加，其中$N$，$L = 12$，$N_{\text{obs}} = 1$，$d_\ell = 30$，$s = 50$。

5 总结

在本综述以及 [47], 中，我们关注这样的场景：目标函数是光滑的，但涉及大量变量且计算代价高昂。如 [4], 所述，近期关于深度神经网络逼近理论的大部分研究都集中于通过模拟证明存在性定理。然而，理论结果与实际性能之间仍存在显著差距（例如， [6, 35]），这促使研究者通过实用存在性理论 （例如， [5]）来弥合这一差距。我们的定理 3 假设存在一个序列（b_j）$_{j\geq 1}$，该序列刻画了目标函数的正则性；这一概念类似于 [4] 中探讨并在 [67, 81]中进一步讨论的全纯函数框架。不同于 [4],所处理的各向异性未知情形——在该情形下，学习算法无需明确依赖于（b_j）$_{j\geq 1}$——我们的方法将关于该序列的知识明确融入格训练点和实用正则化项的设计中。我们从理论上证明，通过将网络参数限制为与目标函数的正则性特征相匹配，可以获得良好的泛化误差界，且该误差界独立于输入维数 s。这种方法符合一些现实应用，例如不确定性量化中的参数化偏微分方程，在这类应用中可以对输入随机场进行显式建模。

[47] 中的数值实验以及本文针对 Swish函数和 ReLU 激活函数开展的新实验表明，我们的定制正则化始终优于标准l_2正则化。

鉴于深度神经网络是用于函数逼近的非线性算法，自然会有人想知道我们的基于格点的周期深度神经网络 (14) 与基于格点的逼近算法 $A^\dagger$ 和 A^* （见 (3) 和 (4)）相比如何。对于 [40], 中的参数化偏微分方程问题，我们知道算法 A^* （以及 $A^\dagger$）

在采用范数 (9) 的周期函数空间设定 (b) 下，其逼近误差收敛率为 L_2，其中 $N^{-r/2}$，并且 $r = 1/p^* - 1/2$。另一方面，我们的基于格点的周期深度神经网络的定理 3(b)(c) 达到 $N^{-r/2}$，其中 $r = 1/p^* - 1/2$ 或 $r = 1/p^*$，但还需加入训练误差 tol 以取得平衡。未来一个有趣的研究方向，是针对 [40]中讨论的同一偏微分方程问题，在实际应用中比较我们的基于格点的周期深度神经网络与这些方法的性能。

致谢

我们感谢澳大利亚研究理事会（DP240100769）和弗兰德斯研究基金会（FWO G091920N）提供的资金支持。

附录

定理 3(a) 的证明。证明与 [57] 中的论证基本一致。从（17）出发，对于每个分量 $1 \le p \le N_{\text{obs}}$，使用定制的随机移位格点规则，我们对所有 $\lambda \in (\frac{1}{2}, 1]$，

$$\text{r.m.s. } e_N^{\text{wor}}(z) \left\| \left(G(\cdot)_p - G_\theta^{[L]}(\cdot)_p\right)^2 \right\|_{\mathcal{W}_{1,\boldsymbol{\gamma}}} \le \left(\frac{2\,(16\,C^4)^\lambda\, C_{s,\boldsymbol{\gamma},\lambda}}{N} \right)^{\frac{1}{2\lambda}},$$

with

$$C_{s,\boldsymbol{\gamma},\lambda} := \left(\sum_{\emptyset \ne \mathfrak{u} \subseteq \{1:s\}} \gamma_{\mathfrak{u}}^\lambda \left[\frac{2\zeta(2\lambda)\,2^\lambda}{(2\pi)^{2\lambda}} \right]^{|\mathfrak{u}|} \right) \left(\sum_{\mathfrak{u} \subseteq \{1:s\}} \frac{1}{\gamma_{\mathfrak{u}}} \left((|\mathfrak{u}|+1)! \prod_{j \in \mathfrak{u}} (\kappa\, S_L\, b_j) \right)^2 \right)^\lambda,$$

其中，我们使用了均方根最坏情形误差界（8）和范数界（30），但将 b_j 替换为 $\kappa\, S_L\, b_j$，其中 κ 如（40）所示。

特殊情形 $\kappa\, S_L = 1$ 对应于（27）成立时的理想情形。在此情形下，（34）中的权重 $\gamma_{\mathfrak{u}}$ 使 $C_{s,\ \boldsymbol{\gamma},\ \lambda}$ 最小化，并使两个求和中的项相等。这与 [57, 定理 6.4] 类似。

对于所有 $\kappa\, S_L \ge 1$，我们现在证明，权重 (34) 的选择确保 $C_{s,\boldsymbol{\gamma},\lambda}$ 有界，且其界与 s 无关。将 (34) 代入$C_{s,\boldsymbol{\gamma},\lambda}$，得到

$$\begin{aligned}
C_{s,\boldsymbol{\gamma},\lambda} &= \left(\sum_{\emptyset \ne \mathfrak{u} \subseteq \{1:s\}} [(|\mathfrak{u}|+1)!]^{\frac{2\lambda}{1+\lambda}} \left[\frac{2\zeta(2\lambda)\,2^\lambda}{(2\pi)^{2\lambda}} \right]^{\frac{\lambda|\mathfrak{u}|}{1+\lambda}} \prod_{j \in \mathfrak{u}} b_j^{\frac{2\lambda}{1+\lambda}} \right) \\
&\quad \times \left(\sum_{\mathfrak{u} \subseteq \{1:s\}} [(|\mathfrak{u}|+1)!]^{\frac{2\lambda}{1+\lambda}} \left[\frac{2\zeta(2\lambda)\,2^\lambda}{(2\pi)^{2\lambda}} \right]^{\frac{\lambda|\mathfrak{u}|}{1+\lambda}} (\kappa\, S_L)^{2|\mathfrak{u}|} \prod_{j \in \mathfrak{u}} b_j^{\frac{2\lambda}{1+\lambda}} \right)^\lambda \\
&\le \left(\sum_{|\mathfrak{u}|<\infty} [(|\mathfrak{u}|+1)!]^{\frac{2\lambda}{1+\lambda}} \prod_{j \in \mathfrak{u}} (\varrho\, b_j)^{\frac{2\lambda}{1+\lambda}} \right)^{1+\lambda}, \qquad \varrho := \left[\frac{2\zeta(2\lambda)\,2^\lambda}{(2\pi)^{2\lambda}} \right]^{\frac{1}{2}} (\kappa\, S_L)^{\frac{1+\lambda}{\lambda}}.
\end{aligned}$$

现在考虑任意的 $\alpha_j > 0$，其具体取值稍后指定。由于 $\lambda \in (1/2,\ 1)$，因此 $\frac{2\lambda}{1+\lambda} < 1$，我们可以使用共轭指数为 $\frac{1+\lambda}{2\lambda}$ 和 $\frac{1+\lambda}{1-\lambda}$ 的 H¨older 不等式，如下所示

$$\begin{aligned}
C_{s,\boldsymbol{\gamma},\lambda}^{\frac{1}{1+\lambda}} &\le \sum_{|\mathfrak{u}|<\infty} [(|\mathfrak{u}|+1)!]^{\frac{2\lambda}{1+\lambda}} \prod_{j \in \mathfrak{u}} \alpha_j^{\frac{2\lambda}{1+\lambda}} \prod_{j \in \mathfrak{u}} \left(\frac{\varrho\, b_j}{\alpha_j} \right)^{\frac{2\lambda}{1+\lambda}} \\
&\le \left(\sum_{|\mathfrak{u}|<\infty} (|\mathfrak{u}|+1)! \prod_{j \in \mathfrak{u}} \alpha_j \right)^{\frac{2\lambda}{1+\lambda}} \left(\sum_{|\mathfrak{u}|<\infty} \prod_{j \in \mathfrak{u}} \left(\frac{\varrho\, b_j}{\alpha_j} \right)^{\frac{2\lambda}{1-\lambda}} \right)^{\frac{1-\lambda}{1+\lambda}} \\
&\le \left(\frac{1}{1 - \sum_{j \ge 1} \alpha_j} \right)^{\frac{4\lambda}{1+\lambda}} \exp\left(\frac{1-\lambda}{1+\lambda}\, \varrho^{\frac{2\lambda}{1-\lambda}} \sum_{j \ge 1} \left(\frac{b_j}{\alpha_j} \right)^{\frac{2\lambda}{1-\lambda}} \right),
\end{aligned}$$

其中我们使用了 $\sum_{|\mathfrak{u}|<\infty}(|\mathfrak{u}|+1)!\prod_{j\in\mathfrak{u}}\alpha_j \le (\frac{1}{1-\sum_{j\ge1}\alpha_j})^2$和 $\sum_{|\mathfrak{u}|<\infty}\prod_{j\in\mathfrak{u}}\beta_j \le \exp(\sum_{j\ge1}\beta_j)$，参见 [57,引理 6.3], ，该结论成立的条件是

$$\sum_{j\ge1}\alpha_j<1 \quad \text{and} \quad \sum_{j\ge1}\left(\frac{b_j}{\alpha_j}\right)^{\frac{2\lambda}{1-\lambda}}<\infty.$$

已知 $\sum_{j\ge1}b_j^{p^*}<\infty$，现在我们选择 $\alpha_j=b_j^{p^*}/\vartheta$，其中 $\vartheta>\sum_{j\ge1}b_j^{p^*}$。于是有$\sum_{j\ge1}\alpha_j<1$，而要使第二个求和有限，我们需要

$$\frac{2\lambda}{1-\lambda}(1-p^*)\ge p^* \iff \lambda\ge\frac{p^*}{2-p^*}.$$

为了获得最佳收敛速率，我们希望 $\lambda\in(1/2,\ 1)$ 尽可能小。因此，我们按照 (34) 指定 λ，以确保获得最佳收敛速率，且相应常数与维度 s 无关。证明至此完成。□

定理 3(b) 的证明。证明思路与 [40] 中的论证基本相同。从 (17) 出发，对每个分量 $1\le p\le N_{\text{obs}}$，使用量身定制的格点规则，我们对所有 $\lambda\in(\frac{1}{2\alpha},\ 1]$，

$$e_N^{\text{wor}}(z)\left\|(G(\cdot)_p-G_\theta^{[L]}(\cdot)_p)^2\right\|_{\mathcal{W}_{\alpha,\boldsymbol{\gamma}}}\le\left(\frac{2\,(16\,C^4)^\lambda\,C_{s,\boldsymbol{\gamma},\lambda}}{N}\right)^{\frac{1}{2\lambda}},$$

with

$$C_{s,\boldsymbol{\gamma},\lambda}:=\left(\sum_{\emptyset\ne\mathfrak{u}\subseteq\{1:s\}}\gamma_{\mathfrak{u}}^\lambda\left[\frac{2\zeta(2\alpha\lambda)}{(2\pi)^{2\alpha\lambda}}\right]^{|\mathfrak{u}|}\right)$$
$$\times\left(\sum_{\mathfrak{u}\subseteq\{1:s\}}\frac{(2\pi)^{2\alpha|\mathfrak{u}|}}{\gamma_{\mathfrak{u}}}\left(\sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}}(|\boldsymbol{m}_{\mathfrak{u}}|+1)!\,\mathcal{S}(\boldsymbol{\alpha}_{\mathfrak{u}},\boldsymbol{m}_{\mathfrak{u}})\prod_{j\in\mathfrak{u}}(\kappa\,S_L\,b_j)^{m_j}\right)^2\right)^\lambda,$$

其中 $\mathsf{S}(\alpha_{\mathfrak{u}},\ \boldsymbol{m}_{\mathfrak{u}}):=\prod_{j\in\mathfrak{u}}\mathsf{S}(\alpha,\ m_j)$，这里我们使用了最坏情况误差界 (10) 和范数界 (32)，但将 b_j 替换为 $\kappa\,S_L\,b_j$，其中 κ 如 (40) 所示。与前一个证明一样，我们接下来证明，权重 (36) 的选择使得$C_{s,\ \boldsymbol{\gamma},\ \lambda}$ 可以得到一个与 s 无关的界。

将权重 (36) 的选取简记为

$$\gamma_{\mathfrak{u}}=\varrho^{|\mathfrak{u}|}\sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}}[V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}})]^{\frac{2}{1+\lambda}},\quad \varrho:=\frac{(2\pi)^{2\alpha}}{[2\zeta(2\alpha\lambda)]^{\frac{1}{1+\lambda}}},$$

$$V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}}):=(|\boldsymbol{m}_{\mathfrak{u}}|+1)!\,\mathcal{S}(\boldsymbol{\alpha}_{\mathfrak{u}},\boldsymbol{m}_{\mathfrak{u}})\prod_{j\in\mathfrak{u}}b_j^{m_j},$$

并将其代入$C_{s,\,\boldsymbol{\gamma},\,\lambda}$，我们得到

$$
\begin{aligned}
C_{s,\boldsymbol{\gamma},\lambda} &= \Bigg(\sum_{\emptyset\neq\mathfrak{u}\subseteq\{1:s\}} \Bigg(\varrho^{|\mathfrak{u}|} \sum_{\boldsymbol{m}_\mathfrak{u}\le\boldsymbol{\alpha}_\mathfrak{u}} [V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})]^{\frac{2}{1+\lambda}} \Bigg)^{\lambda} \left[\frac{2\zeta(2\alpha\lambda)}{(2\pi)^{2\alpha\lambda}}\right]^{|\mathfrak{u}|} \Bigg) \\
&\quad \times \Bigg(\sum_{\mathfrak{u}\subseteq\{1:s\}} \frac{(2\pi)^{2\alpha|\mathfrak{u}|}}{\varrho^{|\mathfrak{u}|}\sum_{\boldsymbol{m}_\mathfrak{u}\le\boldsymbol{\alpha}_\mathfrak{u}}[V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})]^{\frac{2}{1+\lambda}}} \Bigg(\sum_{\boldsymbol{m}_\mathfrak{u}\le\boldsymbol{\alpha}_\mathfrak{u}} [V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})]^{\frac{1}{1+\lambda}} [V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})]^{\frac{\lambda}{1+\lambda}} (\kappa S_L)^{|\boldsymbol{m}_\mathfrak{u}|} \Bigg)^2 \Bigg)^{\lambda} \\
&\le \Bigg(\sum_{\mathfrak{u}\subseteq\{1:s\}} [2\zeta(2\alpha\lambda)]^{\frac{|\mathfrak{u}|}{1+\lambda}} \sum_{\boldsymbol{m}_\mathfrak{u}\le\boldsymbol{\alpha}_\mathfrak{u}} [V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})]^{\frac{2\lambda}{1+\lambda}} \Bigg) \\
&\qquad \times \Bigg(\sum_{\mathfrak{u}\subseteq\{1:s\}} [2\zeta(2\alpha\lambda)]^{\frac{|\mathfrak{u}|}{1+\lambda}} \sum_{\boldsymbol{m}_\mathfrak{u}\le\boldsymbol{\alpha}_\mathfrak{u}} [V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})]^{\frac{2\lambda}{1+\lambda}} (\kappa S_L)^{2|\boldsymbol{m}_\mathfrak{u}|} \Bigg)^{\lambda} \\
&\le \Bigg(\sum_{\mathfrak{u}\subseteq\{1:s\}} [2\zeta(2\alpha\lambda)]^{\frac{|\mathfrak{u}|}{1+\lambda}} \sum_{\mathbf{1}_\mathfrak{u}\le\boldsymbol{m}_\mathfrak{u}\le\boldsymbol{\alpha}_\mathfrak{u}} [V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})]^{\frac{2\lambda}{1+\lambda}} (\kappa S_L)^{2|\boldsymbol{m}_\mathfrak{u}|} \Bigg)^{1+\lambda} \\
&\le \Bigg(\sum_{\boldsymbol{m}\le\boldsymbol{\alpha}} \Bigg((|\boldsymbol{m}|+1)! \prod_{j=1}^{s} \widetilde{b}_j^{\,m_j} \Bigg)^{\frac{2\lambda}{1+\lambda}} \Bigg)^{1+\lambda}, \quad \widetilde{b}_j := \mathcal{S}_{\max}(\alpha)\,[2\zeta(2\alpha\lambda)]^{\frac{1}{2\lambda}}\,(\kappa S_L)^{\frac{1+\lambda}{\lambda}}\,b_j,
\end{aligned}
$$

其中 $\mathsf{S}_{\max}(\alpha) := \max_{1\le m\le\alpha}\mathsf{S}(\alpha,m)$。在上面的第一个不等式中，对于第一个因子，我们去掉了条件 $\mathfrak{u}\neq\emptyset$，并对关于 $\boldsymbol{m}_\mathfrak{u}$ 的求和应用了詹森不等式（即，对于 $a_i\ge 0$ 和 $0<\lambda\le 1$，有 $(\sum_i a_i)^\lambda \le \sum_i a_i^\lambda$）；对于第二个因子，我们对关于$\boldsymbol{m}_\mathfrak{u}$ 的内层求和应用了柯西-施瓦茨不等式，然后约去了分母中的求和。在上面的第二个不等式中，我们使用 $\kappa S_L\ge 1$ 合并了上一步得到的两个因子，并且明确加入了条件 $\boldsymbol{m}_\mathfrak{u}\ge\mathbf{1}_\mathfrak{u}$，因为如果任何 $m_j=0$，那么 $\mathsf{S}(\alpha,m_j)=0$，因此 $V_\mathfrak{u}(\boldsymbol{m}_\mathfrak{u})=0$。在上面的第三个不等式中，我们利用 $\mathsf{S}(\alpha,m_j)\le\mathsf{S}_{\max}(\alpha)\le[\mathsf{S}_{\max}(\alpha)]^{m_j}$ 以及 $2\zeta(2\alpha\lambda)\le[2\zeta(2\alpha\lambda)]^{m_j}$，将这些因子移入$\widetilde{\ }$的定义中；然后，我们将关于 $\mathfrak{u}$ 和 $\boldsymbol{m}_\mathfrak{u}\ge\mathbf{1}_\mathfrak{u}$ 的双重求和等价地改写为关于 $\boldsymbol{m}\le\boldsymbol{\alpha}=(\alpha,\ldots,\alpha)\in\mathbb{N}^s$ 的单一求和。

还需证明$C_{s,\,\boldsymbol{\gamma},\,\lambda}$上的最后一个上界可以被一个与 s 无关的量所界定。我们基本沿用 [40, 第 60 页定理 4 的证明5], 中的论证，注意到这里使用的是 $(|\boldsymbol{m}|+1)!\widetilde{\ }$而非 $|\boldsymbol{m}|!$，并且相对于 b_j，b_j的定义中采用了不同的缩放。

$\widetilde{\ }$ $\widetilde{\ }$ $\widetilde{\ }$我们定义序列 $d_j := b_{\lceil j/\alpha\rceil}$，其中 $j\ge 1$，使得 $d_1=\cdots=d_\alpha=b_1$、$d_{\alpha+1}=\cdots=d_{2\alpha}=b_2$，依此类推。于是

$$
\prod_{j=1}^{s}\widetilde{b}_j^{\,m_j} = \underbrace{\widetilde{b}_1\cdots\widetilde{b}_1}_{m_1\text{ factors}}\cdot\underbrace{\widetilde{b}_2\cdots\widetilde{b}_2}_{m_2\text{ factors}}\cdots\underbrace{\widetilde{b}_s\cdots\widetilde{b}_s}_{m_s\text{ factors}} = \prod_{j\in\mathfrak{v}_{\boldsymbol{m}}} d_j, \tag{43}
$$

其中 $\mathfrak{v}_{\boldsymbol{m}} := \{1、\ldots、m_1、\alpha+1、\ldots、\alpha+m_2、\ldots、(s-1)\alpha+1、\ldots、(s-1)\alpha+m_s\}$ 以及 $|\mathfrak{v}_{\boldsymbol{m}}|=|\boldsymbol{m}|$。因此

$$
\begin{aligned}
\sum_{\boldsymbol{m}\le\boldsymbol{\alpha}} \Bigg((|\boldsymbol{m}|+1)! \prod_{j=1}^{s}\widetilde{b}_j^{\,m_j} \Bigg)^{\frac{2\lambda}{1+\lambda}} &\le \sum_{\substack{\mathfrak{v}\subset\mathbb{N}\\ |\mathfrak{v}|<\infty}} \Bigg((|\mathfrak{v}|+1)! \prod_{j\in\mathfrak{v}} d_j \Bigg)^{\frac{2\lambda}{1+\lambda}} \\
&= \sum_{\ell\ge 0} [(\ell+1)!]^{\frac{2\lambda}{1+\lambda}} \sum_{\substack{\mathfrak{v}\subset\mathbb{N}\\ |\mathfrak{v}|=\ell}} \prod_{j\in\mathfrak{v}} d_j^{\frac{2\lambda}{1+\lambda}} \le \sum_{\ell\ge 0} \underbrace{[(\ell+1)!]^{\frac{2\lambda}{1+\lambda}} \frac{1}{\ell!} \Bigg(\sum_{j\ge 1} d_j^{\frac{2\lambda}{1+\lambda}} \Bigg)^{\ell}}_{=:A_\ell}.
\end{aligned} \tag{44}
$$

$^{2\lambda}$ — $^{2\lambda}$最后一个不等式成立，是因为$(\sum_{j\ge1} d_j^{1+\lambda})^\ell$包含所有形如 $\prod_{j\in\mathfrak{v}} d_j^{1+\lambda}$ 的乘积，其中 $|\mathfrak{v}|=l$，并且每个这样的项都包含$l!$ 次。

回忆一下，$\sum_{j\ge1} b_j^{p^*} < \infty$。在式（36）中，我们对 $\lambda := \frac{p^*}{2-p^*}$ 的选择给出 $\frac{2\lambda}{1+\lambda} = p^*$，因此

$$\sum_{j\ge1} d_j^{\frac{2\lambda}{1+\lambda}} = \sum_{j\ge1} d_j^{p^*} = \alpha \sum_{j\ge1} \widetilde{b}_j^{p^*} = \alpha\,[\mathcal{S}_{\max}(\alpha)]^{p^*}\,[2\zeta(2\alpha\lambda)]^{\frac{1}{1+\lambda}}\,(\kappa\,S_L)^2 \sum_{j\ge1} b_j^{p^*} < \infty,$$

前提是 $2\alpha\lambda > 1$，这等价于 $\alpha > \frac{1}{p^*} - \frac{1}{2}$。这证明了我们将 α 的整数值选为 $= \lfloor \frac{1}{p^*} - \frac{1}{2} + 1 \rfloor = \lfloor \frac{1}{p^*} + \frac{1}{2} \rfloor$ 的合理性。

最后，根据比值判别法，(44) 中对 l 的最后一个求和是有限的，因为

$$\lim_{\ell\to\infty} \frac{A_{\ell+1}}{A_\ell} = \lim_{\ell\to\infty} \frac{(\ell+2)^{p^*}}{\ell+1} \sum_{j\ge1} d_j^{p^*} \le \lim_{\ell\to\infty} \frac{(\ell+1)^{p^*}+1}{\ell+1} \sum_{j\ge1} d_j^{p^*} = 0.$$

因此，我们得出结论：$C_{s,\,\gamma,\,\lambda}$ 有界，且其界与 s 无关。 □

定理3(c)的证明。证明与 [41] 中的论证非常相似。从 (17) 出发，对于每个分量 $1 \le p \le N_{\text{obs}}$，使用定制格点规则，我们对所有 $\lambda \in (\frac{1}{\alpha},\ 1]$，

$$e_N^{\text{wor}}(\boldsymbol{z}) \left\| (G(\cdot)_p - G_\theta^{[L]}(\cdot)_p)^2 \right\|_{\mathcal{W}_{\alpha,\boldsymbol{\gamma}}} \le \left(\frac{2\,(4\,C^2)^\lambda\,C_{s,\boldsymbol{\gamma},\lambda}}{N} \right)^{\frac{1}{\lambda}},$$

with

$$C_{s,\boldsymbol{\gamma},\lambda} := \left(\sum_{\emptyset\ne\mathfrak{u}\subseteq\{1:s\}} \gamma_{\mathfrak{u}}^\lambda \left[\frac{2\zeta(\alpha\lambda)}{(2\pi)^{\alpha\lambda}} \right]^{|\mathfrak{u}|} \right) \times \left(\max_{\mathfrak{u}\subseteq\{1:s\}} \frac{(2\pi)^{\alpha|\mathfrak{u}|}}{\gamma_{\mathfrak{u}}} \sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}} (|\boldsymbol{m}_{\mathfrak{u}}|+1)!\,\mathcal{S}(\boldsymbol{\alpha}_{\mathfrak{u}},\boldsymbol{m}_{\mathfrak{u}}) \prod_{j\in\mathfrak{u}} (\kappa\,S_L\,b_j)^{m_j} \right)^\lambda,$$

这里我们使用了最坏情况误差界 (12) 和范数界 (33)，但将 b_j 替换为 $\kappa\,S_L\,b_j$，其中 κ 如 (40) 所示。

与定理3(a)和定理 3(b)不同，此处权重 $\gamma_{\mathfrak{u}}$ 的选取方式使得在 $\kappa\,S_L = 1$ 的情况下，集合 $\mathfrak{u}$ 上的最大值恰好为 1。将权重的选取方式（38）简记为

$$\gamma_{\mathfrak{u}} = (2\pi)^{\alpha|\mathfrak{u}|} \sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}} V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}}), \quad V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}}) := (|\boldsymbol{m}_{\mathfrak{u}}|+1)!\,\mathcal{S}(\boldsymbol{\alpha}_{\mathfrak{u}},\boldsymbol{m}_{\mathfrak{u}}) \prod_{j\in\mathfrak{u}} b_j^{m_j},$$

并将其代入 $C_{s,\,\boldsymbol{\gamma},\,\lambda}$，我们得到

$$C_{s,\boldsymbol{\gamma},\lambda} = \left(\sum_{\emptyset\ne\mathfrak{u}\subseteq\{1:s\}} \left(\sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}} V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}}) \right)^\lambda [2\zeta(\alpha\lambda)]^{|\mathfrak{u}|} \right) \times \left(\max_{\mathfrak{u}\subseteq\{1:s\}} \frac{1}{\sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}} V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}})} \sum_{\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}} V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}})\,(\kappa\,S_L)^{|\boldsymbol{m}_{\mathfrak{u}}|} \right)^\lambda.$$

注意，在对 $\boldsymbol{m}_{\mathfrak{u}}$ 求和时，我们有 $\boldsymbol{m}_{\mathfrak{u}} \ge \mathbf{1}_{\mathfrak{u}}$；否则，由于 $\mathsf{S}(\alpha,\ 0) = 0$，$V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}}) = 0$。当 $\kappa\,S_L = 1$ 时，$\mathfrak{u}$ 上的最大值恰好为 1，因此可以忽略第二个因子。利用詹森不等式，第一个因子中对 $\mathfrak{u}$ 的求和可以由以下各项界定

$$C_{s,\boldsymbol{\gamma},\lambda} \le \sum_{\mathfrak{u}\subseteq\{1:s\}} \sum_{\mathbf{1}_{\mathfrak{u}}\le\boldsymbol{m}_{\mathfrak{u}}\le\boldsymbol{\alpha}_{\mathfrak{u}}} [V_{\mathfrak{u}}(\boldsymbol{m}_{\mathfrak{u}})]^\lambda\,[2\zeta(\alpha\lambda)]^{|\mathfrak{u}|} \le \sum_{\boldsymbol{m}\le\boldsymbol{\alpha}} \left((|\boldsymbol{m}|+1)! \prod_{j=1}^s \widetilde{b}_j^{m_j} \right)^\lambda \le \sum_{\ell\ge0} [(\ell+1)!]^\lambda\,\frac{1}{\ell!} \left(\sum_{j\ge1} d_j^\lambda \right)^\ell,$$

其中现在 $\widetilde{b}_j := \mathrm{S}_{\max}(\alpha)\ [2\zeta(\alpha\lambda)]^{\frac{1}{\lambda}}\, b_j$，而 d_j 的定义与 (43)–(44) 中相同，但使用了 b_j 的新定义$\widetilde{\ }$。现在取 $\lambda = p^*$，得到

$$\sum_{j\ge 1} d_j^{\lambda} = \sum_{j\ge 1} d_j^{p^*} = \alpha \sum_{j\ge 1} \widetilde{b}_j^{p^*} = \alpha\,[\mathcal{S}_{\max}(\alpha)]^{p^*}\,[2\zeta(\alpha\lambda)] \sum_{j\ge 1} b_j^{p^*} < \infty,$$

在 $\alpha\lambda > 1$ 的前提下，这就说明了在 (38) 中选择 $\alpha := \lfloor \frac{1}{p^*} \rfloor + 1$ 的合理性。随后，利用定理 3(b) 证明最后一部分中的相同论证可得，$C_{s,\ \gamma,\ \lambda}$ 有界，且其界与 s 无关。

当 $\kappa S_L > 1$ 时，由于 $\boldsymbol{m}_{\mathfrak{u}} \ge \mathbf{1}_{\mathfrak{u}}$，对 $\mathfrak{u}$ 取最大值时，其中的每个因子都至少为 $(\kappa S_L)^{|\mathfrak{u}|}$，且至多为 $(\kappa S_L)^{\alpha|\mathfrak{u}|}$。因此，$C_{s,\ \gamma,\ \lambda}$ 中包含这一额外因子，其值介于 $(\kappa S_L)^s$ 和 $(\kappa S_L)^{\alpha s}$ 之间；遗憾的是，该因子随 s 指数增长。□

参考文献

[1] Achtsis, N., Cools, R., Nuyens, R.：Heston 模型下障碍期权定价的条件采样。载于：Dick, J., Kuo, F.Y., Peters, G., Sloan, I.H.（编），2012 年蒙特卡罗与拟蒙特卡罗方法，第 253–269 页。施普林格（2013）[2] Achtsis, N., Cools, R., Nuyens, R.：LT 方法下障碍期权定价的条件采样。《SIAM 金融数学杂志》**4**，327–352（2013）[3] Acworth, P., Broadie, M., Glasserman, P.：用于期权定价的若干蒙特卡罗与准蒙特卡罗技术比较。载于：Hellekalek, P., Larcher, G., Niederreiter, H., Zinterhof, P.（编），1996 年蒙特卡罗与拟蒙特卡罗方法，第 1–18 页。Springer 出版社，柏林（1998）[4] Adcock, B., Brugiapaglia, S., Dexter, N., Moraga, S.：高维光滑函数学习：从稀疏多项式到深度神经网络。《数值分析手册》**25**，1–52（2024）[5] Adcock, B., Brugiapaglia, S., Dexter, N., Moraga, S.：通过深度神经网络近乎最优地学习巴拿赫空间值高维函数。《神经网络杂志》**181**，106761（2025）[6] Adcock, B., Dexter, N.：深度神经网络函数逼近的理论与实践之间的差距。《SIAM 数学与数据科学杂志》**3**，624–655（2021）[7] Bartel, F., K¨ammerer, L., Potts, D., Ullrich, T.：基于欠采样求积点处的函数值重构函数。《计算数学》**93**，785–809（2024）[8] Buchholz, A., Wenzel, F., Mandt, S.：拟蒙特卡罗变分推断。《机器学习研究论文集》**80**，668–677（2018）[9] Byrenheid, G., K¨ammerer, L., Ullrich, T., Volkmer, T.：混合混合光滑性空间中秩1格点采样的紧误差界。《Numerische Mathematik》**136**，993–1034（2017）[10] Caflisch, R.E., Morokoff, W.J., Owen, A.B.：利用布朗桥降低有效维数，对抵押贷款支持证券进行估值。《计算金融》**1**，27–46（1997）[11] Cools, R., Kuo, F.Y., Nuyens, D.：构造用于多元积分的嵌入式格点规则。《SIAM 科学计算杂志》**28**，2162–2188（2006）[12] Cools, R., Kuo, F.Y., Nuyens, D., Sloan, I.H.：具有一般权重参数的周期函数空间中多元逼近的格算法。载于：Brenner, S.C., Shparlinski, I., Shu, C.-W., Szyld, D.（编），庆祝《计算数学》创刊 75 周年，《当代数学》，第 754 卷，

pp。93–113。AMS（2020年）

[13] Cools, R., Kuo, F.Y., Nuyens, D., Sloan, I.H.：基于 POD 和 SPOD 权重的多元逼近格算法的快速逐分量构造。《Mathematics of Computation》**90**，787–812（2021）[14] Cools, R., Kuo, F.Y., Nuyens, D., Suryanarayana, G.：用于多元非周期函数积分与逼近的帐篷变换格规则。《Journal of Complexity》，**36**，166–181（2016）[15] Dammertz, S., Keller, A.：基于秩-1格的图像合成。载于：Keller, A., Heinrich, S., Nieder- reiter, H.（编）《2006年蒙特卡罗与拟蒙特卡罗方法》。施普林格，柏林，海德堡，第217–236页（2008）[16] Dammertz, H., Keller, A., Dammertz, S.：基于秩-1格的模拟。载于：Keller, A., Heinrich, S., Niederreiter, H.（编）《2006年蒙特卡罗与拟蒙特卡罗方法》。施普林格，柏林，海德堡，第205–216页（2008）。[17] Dick, J., Feischl M.：一种用于机器学习的准蒙特卡罗数据压缩算法。《Journal of Complexity》**67**，101587（2021）[18] Dick, J., Goda, T., Suzuki, K.：逐分量构造随机秩1格规则，使其达到近乎最优的随机误差率。《Mathematics of Computation》**91**，2771–2801（2022）[19] Dick, J., Kritzer, P., Pillichshammer, F.：《格点规则：数值积分、逼近与差异度》。施普林格（2022）[20] Dick, J., Kuo, F.Y., Le Gia, Q.T., Nuyens, D., Schwab, Ch.：用于参数算子方程的高阶准蒙特卡罗 Galerkin 离散化。《SIAM 数值分析杂志》**52**，2676–2702（2014）[21] Dick, J., Kuo, F.Y., Sloan, I.H.：高维积分：准蒙特卡罗之道。《Acta Numerica》**22**，133–288（2013）[22] Dick, J., Nuyens, D., Pillichshammer, F.：用于非周期光滑被积函数的格点规则。《Numerische Mathematik》**126**，259–291（2014）[23] Dick, J., Pillichshammer, F.：《数字网与数字序列：差异度理论与准蒙特卡罗积分》。剑桥大学出版社（2010）[24] Dick, J., Sloan, I.H., Wang, X., Wo´zniakowski, H.：一般权重加权 Korobov 空间中的优良格点规则。《Numerische Mathematik》**103**，63–97（2006）[25] Fu, F., Wang, X.：用于求解偏微分方程的准蒙特卡罗深度学习算法的收敛性分析。《Numerical Mathematics: Theory, Methods and Applications》**16**，668–700（2023）[26] Gilbert, A.D., Graham, I.G., Kuo, F.Y., Scheichl, R., Sloan, I.H.：具有随机系数的椭圆特征值问题的准蒙特卡罗方法分析。《Numerische Mathematik》**142**，863–915（2019）[27] Gilbert, A.D., Kuo, F.Y., Sloan, I.H., Srikumar, A.：期权定价与密度估计准蒙特卡罗规则的理论与构造。载于：Hinrichs, A., Kritzer, P., Pillichshammer, F.（编）《2022年蒙特卡罗与拟蒙特卡罗方法》，第277–295页。施普林格（2024）[28] Goda, T., L'Ecuyer, P.：适用于未指定光滑性函数空间与一般权重的免构造中位数准蒙特卡罗规则。《SIAM 科学计算杂志》**44**，A2765–A2788（2022）[29] Goodfellow, I., Bengio, Y., Courville, A.：《深度学习》。麻省理工学院出版社，马萨诸塞州剑桥（2016）[30] G¨ottlich, S., Heieck,J., Neuenkirch, A.：使用低差异度点进行机器学习数据压缩：实验比较。《数学工业杂志》**15**，1（2025）

[31] Graham, I.G., Kuo, F.Y., Nichols, J.A., Scheichl, R., Schwab, Ch., Sloan, I.H.：用于具有对数正态随机系数的椭圆型偏微分方程的拟蒙特卡罗有限元方法。《Numerische Mathematik》**131**, 329–368（2015）[32] Graham, I.G., Kuo, F.Y., Nuyens, D., Scheichl, R., Sloan, I.H.：用于具有随机系数的椭圆型偏微分方程的拟蒙特卡罗方法及其应用。《Journal of Computational Physics》**230**, 3668–3694（2011）[33] Graham, I.G., Kuo, F.Y., Nuyens, D., Scheichl, R., Sloan, I.H.：基于准蒙特卡罗的循环嵌入：具有对数正态系数的椭圆型偏微分方程分析。《Numerische Mathematik》**140**, 479–511（2018）[34] Grohs, P., Kutyniok, G.（编）：《深度学习的数学方面》。剑桥大学出版社（2002）[35] Grohs, P., Voigtlaender, F.：通过神经网络逼近空间的采样复杂度界证明深度学习中的理论与实践差距。《Foundations of Computational Mathematics》**24**, 1085–1143（2024）[36] Guth, P.A., Kaarnioja, V., Kuo, F.Y., Schillings, C., Sloan, I.H.：利用拟蒙特卡罗积分处理不确定性下具有熵风险度量的抛物型偏微分方程约束最优控制。《Numerische Mathematik》**156**, 565–608（2024）[37] Hakula, H., Harbrecht, H., Kaarnioja, V., Kuo, F. Y., Sloan, I.H.：利用周期随机变量对随机区域进行不确定性量化。《Numerische Mathematik》**156**, 273–317（2024）[38] Hickernell, F.J.：格点规则：其测度表现如何？载于：Hellekalek, P., Larcher, G.（编）《随机点集与拟随机点集》，第109–166页。施普林格（1998年）[39] Hickernell, F.J.：格点求积规则的 $O(N^{-2+\epsilon})$收敛性。载于：Fang, K.T., Hickernell, F.J., Niederreiter, H.（编）《2000年蒙特卡罗与拟蒙特卡罗方法》，第274–289页。施普林格，柏林（2002年）[40] Kaarnioja, V., Kazashi, Y., Kuo, F.Y., Nobile, F., Sloan, I.H.：基于周期核的格点插值实现快速逼近及其在不确定性量化中的应用。《Numerische Mathematik》**150**, 33–77（2022）[41] Kaarnioja, V., Kuo, F.Y., Sloan, I.H.：利用周期随机变量进行不确定性量化。《SIAM 数值分析杂志》**58**, 1068–1091（2020）[42] Kaarnioja, V., Kuo, F.Y., Sloan, I.H.：极高维参数化偏微分方程的基于格的核逼近与巧合权重。载于：Hinrichs, A., Kritzer, P., Pillichshammer, F.（编）《2022年蒙特卡罗与拟蒙特卡罗方法》，第81–103页。施普林格（2024年）[43] K¨ammerer, L.：从沿秩1格点采样的数据重构双曲交叉三角多项式。《SIAM 数值分析杂志》**51**, 2773–2796（2013）[44] K¨ammerer, L., Potts, D., Volkmer, T.：基于秩1格点采样，用三角多项式逼近多元周期函数。《复杂性杂志》**31** 543–576（2015）[45] Keller, A.：计算机图形学的迷思。载于：Niederreiter, H.（编）《2004年蒙特卡罗与拟蒙特卡罗方法》，第217–243页。施普林格，柏林（2006）。[46] Keller, A., Dahm, K.：积分方程与机器学习。《模拟中的数学与计算机》**161**, 2–12（2019）[47] Keller, A., Kuo, F.Y., Nuyens, D., Sloan, I.H.：深度神经网络的正则性与定制正则化及其在不确定性量化中参数化偏微分方程上的应用。即将在《数学计算》发表（2026）（http://arxiv.org/abs/2502.12496）

[48] Keller, A., Van keirsbilck, M.：由低差异序列生成的人工神经网络。载于：Keller, A.（编）《2020年蒙特卡罗与拟蒙特卡罗方法》，第291–311页。施普林格（2022年）[49] Keller, A., W¨achter, C., Binder, N.：用于图形软件的拟蒙特卡罗算法（及其他用途）。载于：Hinrichs, A., Kritzer, P., Pillichshammer, F.（编）《2022年蒙特卡罗与拟蒙特卡罗方法》，第373–391页。施普林格（2024年）[50] Kingma, D.P., Ba, J.：Adam：一种随机优化方法。载于：第三届国际学习表征会议论文集，ICLR（2015）[51] Kritzer, P., Kuo, F.Y., Nuyens, D., Ullrich, M.：具有随机 n 的格点规则可实现近乎最优且与维度无关的 O($n^{-\alpha-1/2}$) 误差。《逼近论杂志》**240**, 96–113（2019）[52] Kuo, F.Y., Migliorati, G., Nobile, F., Nuyens, D.：利用秩1格点进行函数积分、重构与逼近。《数学计算》**90**, 1861–1897（2021）[53] Kuo, F.Y., Mo, W., Nuyens, D.：构造用于多元函数逼近的嵌入式格基算法，点数为合数。《构造逼近》**61**, 81–113（2025年）[54] Kuo, F.Y., Mo, W., Nuyens, D., Sloan, I.H., Srikumar, A.：格基核逼近的两种搜索准则比较。载于：Hinrichs, A., Kritzer, P., Pillichshammer, F.（编）《2022年蒙特卡罗与拟蒙特卡罗方法》，第413–429页。施普林格（2024年）[55] Kuo, F.Y., Nuyens, D.：拟蒙特卡罗方法在具有随机扩散系数的椭圆型偏微分方程中的应用——分析与实现综述。《Foundations of Computational Mathematics》**16**, 1631–1696（2016）[56] Kuo, F.Y., Nuyens, D., Wilkes, L.：用于高维积分的随机素数—固定向量随机化格基算法。《复杂性杂志》**79**, 101785（2023）[57] Kuo, F.Y., Schwab, Ch., Sloan, I.H.：用于一类具有随机系数的椭圆型偏微分方程的拟蒙特卡罗有限元方法。《SIAM 数值分析杂志》**50**, 3351–3374（2012）[58] Kuo, F.Y., Sloan, I.H.：摆脱维数灾难。《美国数学学会通知》**52**, 1320–1328（2005年）[59] Kuo, F.Y., Sloan, I.H., Wo´zniakowski, H.：最坏情形设置下的多元逼近格点规则。载于：Niederreiter, H., Talay, D.（编）《2004年蒙特卡罗与拟蒙特卡罗方法》，第289–330页。施普林格，柏林（2006年）[60] Kuo, F.Y., Wasilkowski, G.W., Wo´zniakowski, H.：最坏情形设置下用于多元 L_∞ 逼近的格算法。《构造逼近》**30**, 475–493（2009年）[61] LeCun, Y., Bengio, Y., Hinton, G.：深度学习。《自然》**521**, 436–444（2015）[62] Lemieux, C.：《蒙特卡罗与拟蒙特卡罗抽样》。施普林格统计学丛书，纽约（2009年）[63] Leobacher, G., Pillichshammer, F.：《拟蒙特卡罗积分与应用导论》。施普林格（2014）[64] Liu, S.：朗之万拟蒙特卡罗。NeurIPS'23 论文集，文章 3292，75338–75353（2023）[65] Liu, S., Owen, A.B.：变分贝叶斯中的拟蒙特卡罗拟牛顿方法。《机器学习研究杂志》**22**, 1–23（2021）

[66] Lye, K.O., Mishra, S., Ray, D.：计算流体力学中的深度学习观测量。《J. Computat. Phys.》**410**109339 (2020)[67] Longo, M., Mishfra, S., Rusch, T.K., Schwab, Ch.：深度神经网络的高阶准蒙特卡洛训练。《SIAM J. Sci. Comput.》**43**, A3938–A3966 (2021)[68] Minai, A.A., Williams, R.D.：关于 sigmoid 函数的导数。《神经网络》**6**, 845–853 (1993)[69] Mishfra, S., Rusch, T.K.：通过使用低差异序列训练来提高深度学习算法的准确性。《SIAM J. Numer. Anal.》**59**: 1811–1834 (2021)[70] Mohri, A.R.M., Talwalkar, A.：《机器学习基础》。麻省理工学院出版社，马萨诸塞州剑桥 (2018)[71] M¨uller, T., Evans, A., Schied, C., Keller, A.：采用多分辨率哈希编码的即时神经图形基元。《ACM Trans. Graph.》**41** (SIGGRAPH 2022), 102:1–15 (2022)[72] M¨uller, T., Rousselle, F., Nov´ak, J., Keller, A.：用于路径追踪的实时神经辐射缓存。《ACM Trans. Graph.》**40** (SIGGRAPH 2021), 36:1–16 (2021)[73] Nichols, J.A., Kuo, F.Y.：快速逐分量构造随机平移格点规则，使其对加权空间中无界被积函数达到 $O(n^{-1+\delta})$ 收敛率，其中 $\mathbb{R}^s$ 配备 POD 权重。《J. Complexity》**30**, 444–468 (2014)[74] Niederreiter, H.：《随机数生成与准蒙特卡洛方法》。工业与应用数学学会，费城 (1992)[75] Nuyens, D., Cools, R.：在平移不变再生核希尔伯特空间中，快速逐分量构造秩1格点规则的算法。《Math. Comp.》**75**, 903–920 (2006)[76] Nuyens, D., Suzuki Y.：用于 $\mathbb{R}^d$ 上积分的缩放格点规则，通过对周期函数空间的正交投影进行误差分析，实现更高阶收敛。《Math. Comp.》**92**, 307–347 (2023)[77] Olver, F.W.J., Olde Daalhuis, A.B., Lozier, D.W., Schneider, B.I., Boisvert, R.F., Clark, C.W., Miller, B.R., Saunders, B.V., Cohl, H.S., McClain, M.A.（编）：NIST数学函数数字图书馆。2024年6月15日发布的1.2.1版, http://dlmf.nist.gov/[78] Paskov, S.H., Traub, J.F.：更快速的金融衍生品估值。《J. Portfolio Management》**22**, 113–120 (1995)[79] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.：PyTorch：一款命令式风格的高性能深度学习库。载于：神经信息处理系统进展32，第8024–8035页。Curran Associates公司 (2019)[80] Potts, D., Volkmer, T.：基于秩1格点采样的稀疏高维快速傅里叶变换。《Appl. Comput. Harmon. Anal.》**41**, 713–748 (2016)[81] Schwab, Ch., Zech, J.：高维深度学习：用于不确定性量化中广义多项式混沌展开的神经网络表达率。《Anal. Appl. (Singap.)》**17**, 19–55 (2019)[82] Shalev-Shwartz, S., Ben-David, S.：《理解机器学习：从理论到算法》。剑桥大学出版社 (2014)

[83] Sitzmann, V., Martel, J., Bergman, A., Lindell, D., Wetzstein, G.：采用周期激活函数的隐式神经表示。NeurIPS论文集 (2020)[84] Sloan, I.H., Joe, S.：《多重积分的格点规则》。牛津大学出版社 (1994)[85] Sloan, I.H., Kaarnioja, V.：加倍收敛速率：改进正交投影的误差界及其在插值中的应用。《BIT Numer. Math.》**65**, 10 (2025)[86]Sloan, I.H., Wo´zniakowski, H.：拟蒙特卡罗算法何时适用于高维积分？《J. Complexity》**14**, 1–33 (1998)

[87] Vaidyanathan, K., Salvi, M., Wronski, B., Akenine-M¨oller, T., Ebelin, P., Lefohn, A.：材质纹理的随机访问神经压缩。ACM Trans. Graph.**42**（SIGGRAPH 2023），88:1–25（2023年）

[88] Wang, X., Sloan, I.H.：为什么高维金融问题通常具有较低的有效维数。SIAM J. Sci. Comput. **27**，159–183（2005年）

[89] Wang, X., Sloan, I.H.：布朗桥与主成分分析：旨在消除维数灾难。IMA J. Numer. Anal. **27**，631–654（2007年）

[90] Wang, X., Sloan, I.H.：金融工程中的准蒙特卡罗方法：等价原理与降维。Oper.Res. **59**，80–95（2011年）

[91] Xiao, J., Fu, F., Wang, X.：基于随机化准蒙特卡罗方法求解线性柯尔莫哥洛夫偏微分方程的深度学习。J. Comput. Appl. Math.**451**，116088（2024年）

[92] Zeng, X.Y., Leung, K.T., Hickernell, F.J.：利用格设计分析周期问题的样条误差。载于：Niederreiterm H., Talay, D.（编），《2004年蒙特卡罗与拟蒙特卡罗方法》，第501–514页。施普林格，柏林（2006年）

[93] Zeng, X.Y., Kritzer, P., Hickernell, F.J.：使用积分格和数字网的样条方法。Constr. Approx. **30**，529–555（2009年）

作者地址

Alexander Keller英伟达，柏林，德国。电子邮箱：akeller@nvidia.com**Frances Y.Kuo**悉尼新南威尔士大学数学与统计学院，悉尼，澳大利亚。电子邮箱：f.kuo@unsw.edu.au**Dirk Nuyens**鲁汶大学计算机科学系，鲁汶，比利时。电子邮箱：dirk.nuyens@kuleuven.be**Ian H.Sloan**悉尼新南威尔士大学数学与统计学院，悉尼，澳大利亚。电子邮箱：i.sloan@unsw.edu.au