神经网络与样条：数值挤出机设计的进展

Jaewook Lee[a,*]、Sebastian Hube[b]、Stefanie Elgeti[a,b]

[a]轻量化设计与结构生物力学研究所（*E317*），*TU Wien*，*GumpendorferStr. 7*，*A-1060* 维也纳，奥地利[b]技术系统计算分析教席（*CATS*），亚琛工业大学，*Schinkelstr. 2*，*52062* 亚琛，德国

arXiv:2302.04998v1 [cs.CE] 10 Feb 2023

摘要

我们提出了一种将神经网络应用于设计改进型单螺杆挤出机混合元件的新方法。具体而言，我们提出在数值形状优化中使用神经网络对几何形状进行参数化。几何参数化对于实现高效的形状优化至关重要，因为它能够仅使用少量设计变量来优化复杂形状。近期方法通常结合使用计算机辅助设计（CAD）数据与基于样条的方法，其中样条的控制点充当设计变量。因此，这些方法依赖于与人类设计者所指定的相同设计变量。虽然这种选择较为便利，但它要么将设计限制在对给定初始设计特征进行小幅修改的范围内——实际上阻止了拓扑变化——要么产生数量多得不理想的设计变量。在本研究中，我们不再采用 CAD 和基于样条的方法，而是利用生成式神经网络构建一个特征丰富但低维的优化空间。使用神经网络进行几何参数化，使现有最先进的方法得到扩展：由此产生的设计空间不再局限于用户预先规定的对特定基准形状的修改。相反，在同一优化空间内，我们可以在看似无关的设计之间进行插值和探索。为展示这一新方法的性能，我们将所开发的形状参数化方法整合到用于塑料挤出动态混合元件的数值设计框架中。最后，我们在与当前基于自由形变的方法进行竞争性对比的环境中检验这一新方法，并证明该方法即使处于早期阶段也已展现出良好性能。

关键词：形状优化，单螺杆挤出机，神经网络，混合，过滤器，几何参数化

1. 引言

高性能计算机的普及和神经网络的出现，推动了现代数值设计的发展。神经网络在图像识别等领域已得到广泛应用，但其进一步推动聚合物加工发展的潜力尚未得到充分发掘。本研究旨在为实现这一目标作出贡献。我们将深度神经网络与成熟的形状优化方法相结合，通过一种新颖的数值设计来提升单螺杆挤出机的混合能力。

在许多聚合物加工环节中，基于螺杆的机器发挥着关键作用。例如，螺杆可用作塑化机，为注塑成型制备聚合物熔体；也可用于型材挤出中的挤出机。为求简便，下文将这类基于螺杆的机器统称为挤出机。在挤出机的众多类型中，单螺杆挤出机（SSE）因经济优势和操作简便而尤为普及。经济因素也推动着当前提高产量的努力。提高产量的方法是使用高速旋转的挤出机。然而，当前单螺杆挤出机混合能力较差，限制了相关进展，因此，提升混合能力成为一项研究课题 [1, 2, 3, 4, 5, 6]。

本文特别关注能够缓解这一局限的改进型混合元件。基于解析推导、实验和基于仿真的研究，已有多种改进混合元件的方法被提出。下文将回顾这三个领域的最新进展。随后，我们将概述神经网络领域的相关进展，并最终阐述在混合元件数值设计中使用神经网络的理由。

*通讯作者电子邮箱：jaewook.lee@tuwien.ac.at (Jaewook Lee), hube@cats.rwth-aachen.de(Sebastian Hube), stefanie.elgeti@tuwien.ac.at (Stefanie Elgeti)

提交至 *Engineering With Computers* 的预印本

由于压力和温度都很高，分析挤出机内部的流动是一项艰巨的任务。因此，早期研究主要聚焦于解析模型和几何形状较为简单的螺杆段，例如计量段 [7]。实验补充了这些理论推导，并使分析得以扩展到更复杂的螺杆段。据Gale所述，典型配置依赖于固化熔体的显微照片 [2] ，借此研究流道或挤出物的横截面。这类流道显微照片的一个实例是Kim和Kwon通过冷螺杆挤出对屏障螺杆开展的开创性研究 [8]。除了研究固化熔体流之外，偶尔也有研究尝试分析挤出机实际运行时的熔体流动，例如Wong 等人 [9]。尽管这类实验取得了巨大成功，但其一项常见局限是只关注单一操作条件。相比之下，数值分析能够以大幅降低的成本研究不同设计和操作点，因此得到广泛应用。下文将概述这类数值分析。

一个早期案例是 Kim和Kwon 对条纹形成进行的准三维有限元（FE）仿真，该研究探讨了屏障螺棱的影响 [10]。另一个案例是 Domingues 等人的研究，他们针对液-液和固-液体系中的分散混合与分布混合，获得了全局混合指数 [11]。他们采用二维简化模型，仿真域从料斗延伸至计量段，其框架甚至支持设计优化。

早期研究通常忽略混合段，而近年来研究混合器的影响已成为重要课题。Celik 等人采用三维流动模拟结合粒子追踪方法，根据基于变形的指数确定混合程度 [1]。另一个案例是 Marschik 等人的研究，该研究比较了不同 Block-Head混合螺杆在分布混合和分散混合方面的表现 [6]。Roland 等人也报告了一项类似研究，重点考察不同菠萝式混合器的混合能力 [3]。这两项研究都采用了三维非牛顿流动模拟。除了这类旨在数值评估给定螺杆设计的研究外，其他聚合物加工领域也有数值设计的相关报道。例如，Elgeti 等人通过形状优化，力求实现口模平衡并减少口模膨胀 [12, 13]。Gaspar-Cunha和Covas也报告了基于优化的设计，他们调整进料区和压缩区的长度、进料区和计量区的螺杆内径、螺距及螺棱间隙 [14]。Potente和T¨obben近期还开展了一项关于混合元件的研究，建立了剪切段压力-产量和功耗的经验模型，以用于数值设计 [15]。最后， [12] 报告了首种将受形状优化方法启发的手段与用于量化混合的目标函数相结合、以设计混合段的方法 [16]。

然而，上述形状优化有一个共同点：本质上，它们只修改预定义的几何特征。在许多情况下，例如模头或模具设计中，这种做法是可以接受的，因为最终产品的形状与初始形状相近（即形状变化较小）。然而，拓扑灵活的形状参数化能为混合元件设计带来更大的优化收益，因为最优几何形状可能与初始形状大相径庭。可实现的改进促使人们开展几何参数化研究。

成熟的形状参数化方法包括径向基函数（RBF） [17], 曲面参数化（采用贝塞尔曲面 [18], 和曲面样条 [19]）。所有这些方法都可以视为过滤器：它们用少量变量对几何形状进行参数化，但代价是缺乏局部控制。在形状优化中使用曲面样条的方法也见于 [12, 13]。与曲面样条类似，自由形变（FFD）[20] 将待变形物体封装在体样条中，从而可以进一步调整样条，使其更适合高效优化。另一种方法则不将几何形状参数化为过滤器，而是使用计算网格的网格节点作为形状参数 [21]。值得庆幸的是，随着神经网络的出现，具备出色灵活性的新型形状参数化方法应运而生。作为引言的收尾，我们将概述这一领域中最相关的研究成果。

许多神经网络本质上都是分类器。这些神经网络是经过优化（即训练）的非线性算法，用于判断对象之间可能有悖直觉的相似性和差异性，从而将对象区分开来。一个典型的应用场景是利用红绿蓝（RGB）像素数据进行图像识别。不过，神经网络也可以经过训练，对远超 RGB 像素值范围的特征进行分类。例如，风格迁移或纹理合成[22]：生成图像时，除了以复现像素数据为目标，还会结合感知数据。这使图像变换成为可能，即将一幅图像的风格迁移到另一幅图像的主题上。Friedrich等人 [23]首次报告了将这些理念拓展到三维形状的研究。通过比较不同的形状表示，作者发现风格迁移同样适用于形状。

我们的工作尤其受到 Liu等人 [24], 的启发，他们使用一种所谓的变分形状学习器来学习

三维形状的体素表示。此处的学习 是指创建一种所谓的潜在空间，即一个低维、富含特征的嵌入空间，用于表示各种形状并在它们之间进行形变。研究表明，借助潜在表示，不仅可以进行简单的形状插值，还可以将几何特征从一种形状迁移到另一种形状。 [25, 26]也展示了基于体素的形状学习的成功案例。在形状表示方面，广泛采用的方法包括基于点云的方法 [27, 28, 29]，（利用三维点集的坐标），以及基于多边形网格的方法，后者使用模板网格 [30, 31] 或多个网格平面 [32] 。

尽管前文提到的表示方法表明，学习三维形状的嵌入空间是可行的，但每项工作都至少缺少以下一项特性：封闭曲面、灵活的输出分辨率，以及平滑连续的表面细节。近期研究通过学习由连续隐式函数（如有符号距离函数（SDF） [33] 和二值占据函数[34, 35], ）表示的形状，满足了上述特性；这些形状可以从中提取为等值面。

我们利用这一潜在空间丰富的特征表示，降低给定混合元件形状优化问题中优化空间的维度。与近期基于样条的滤波器相比，本方法的重要创新在于，神经网络能够找到一些可能违反直觉的方式，在不受用户定义的设计特征限制的情况下，为一组差异显著的形状建立通用参数化。这种从人类设计者抽象而来的方法能够实现维度较低、灵活性却高得多的形状参数化，这也构成了本文工作的动机。

本文结构如下：首先，我们在第2节概述数值形状优化与样条，由此引出几何过滤器的概念。在此基础上，我们将在第3节说明如何利用神经网络，为形状优化创建合适的几何参数化。在第4节中，我们回顾所使用的软件组件，总结所提出框架的构建模块，并详细说明其与基于样条的形状优化方案之间的具体差异。第5节介绍新方法所得结果，并与当前基于样条的设计进行比较。最后，我们在第6节讨论结果并展望后续发展。

2. 几何滤波器作为形状优化框架的组成部分

下一节将讨论形状参数化，这是数值形状优化框架的构成要素之一。因此，我们首先介绍一般的形状优化问题，随后回顾基于样条的形状参数化。在对形状优化框架作出这一总体介绍后，我们将在第 3 节继续讨论为适配神经网络而需要进行的具体修改。

2.1 数值形状优化框架的构成要素

一般的优化问题可表述为使代价函数 J 最小化，该函数将设计变量 $\boldsymbol{\sigma}$ 与某种输出相关联——此处的输出是特定混合元件（即某种特定设计）所获得的混合能力程度。在形状优化中，这一最小化问题通常需要满足两组约束：（1）不等式条件和等式条件，以及设计变量的有界约束；（2）偏微分方程 (PDEs)，每种设计都必须满足这些方程，才能成为可行解。由此得到以下表述：

$$J : \mathbb{R}^{n_\sigma} \mapsto \mathbb{R} \tag{1a}$$

$$\underset{\boldsymbol{\sigma}\in\Sigma\subset\mathbb{R}^n}{\arg\min}\ J(\boldsymbol{\sigma}) \tag{1b}$$

$$\text{s.t. } \mathbf{F}(\boldsymbol{\sigma}) = \mathbf{0} \quad \text{in } \Omega(\boldsymbol{\sigma}), \tag{1c}$$

$$\sigma_i \geq \sigma_{min,i}, \quad i = 1, ...n_\sigma, \tag{1d}$$

$$\sigma_i \leq \sigma_{max,i}, \quad i = 1, ...n_\sigma. \tag{1e}$$

此处，(1d) 和 (1e) 描述了优化变量 $\boldsymbol{\sigma}$ 的有界约束，而 (1c) 表示控制偏微分方程组。数值求解此类偏微分方程约束 设计问题的一种方法，是交替执行 (1) 形状更新和 (2) 代价函数值的计算。对于所研究的混合元件设计用例，这将得到图 1 所示的计算步骤。首先，我们更新形状（即覆盖混合元件的仿真域）。随后，我们使用修改后的计算域计算流场，并据此推导出

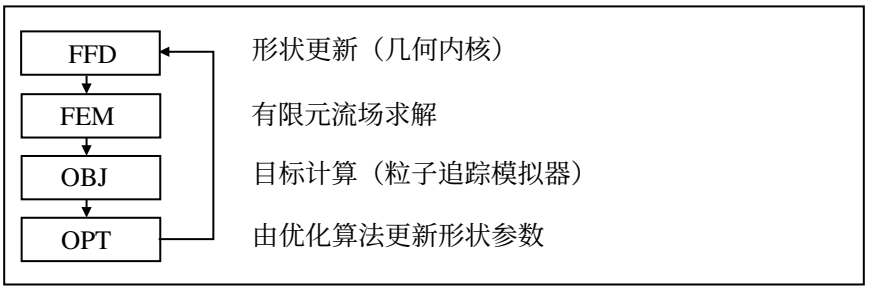


图1：形状优化框架的构成模块。形状由几何内核（如自由形变）进行更新。随后，使用更新后的形状计算流场，并将其作为输入提供给目标计算器。优化算法根据当前设计变量和设计的目标值，计算出优化后的形状参数，并重新启动设计循环，直至满足至少一个设计循环的终止准则。

目标函数（即代价函数）。将代价函数值反馈给优化算法后，设计循环得以闭合；优化算法随后计算更新后的形状。该循环持续进行，直到满足某个终止准则，例如目标值下降幅度达到最小值、达到最大迭代次数或满足其他条件。

2.2. 基于样条的形状参数化

在经典的形状优化框架中，实际的形状参数化或几何滤波通常通过样条实现。因此，下面先概述样条，并说明如何利用它们进行滤波。如需了解 B样条的详细介绍，请参阅 Piegl 和 Tiller 的著作 [19]。之后，我们将详细介绍边界样条和自由形变（FFD）这两种样条参数化的具体应用。

样条属于参数化形状表示这一类。因此，参数空间中的每个坐标都对应物理空间中的一个点。通过一个简单的 B样条曲面，可以更好地理解这种映射，其表达式如下：

$$\mathbf{S}(\xi,\eta) = \sum_{j=1}^{m}\sum_{i=1}^{n} N_{i,r}N_{j,p}(\xi)\,\mathbf{B}_{i,j}, \tag{2}$$

其中，ξ 和 η 表示参数坐标（对于曲面而言有两个参数坐标）；$N_{i,r}$ 表示第一个参数方向上阶数为 r 的插值函数或基函数，$N_{j,p}$ 表示第二个参数方向上阶数为 p 的基函数；最后，$\mathbf{B}$ 表示支撑点或控制点。图2展示了这一概念，并直观呈现了单个控制点对几何形状的影响。控制网格（即由控制点构成的多边形）与 ξ 和 η

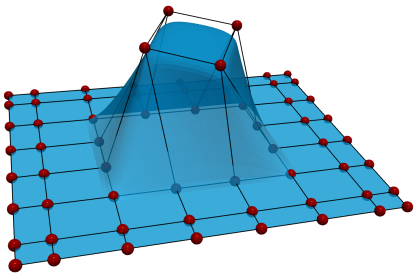


图2：B样条表　　示（蓝色）由双二次B样条的控制点（红色）生成。上方四个控制点经过旋转　　，以展示一种可能的 变形。

方向，并且任意参数坐标（在样条的参数边界内）都映射到蓝色形状上的一个点。因此，样条映射能够用数量固定且通常较少的控制点，控制任意数量的参数点。用少量控制点控制大量点，正是样条滤波的基本思想。

可以通过多种方式利用样条获得几何参数化。如图2所示，其中一种方式是将B样条用作边界表示。这种基于样条的边界表示在CAD中十分常见。使用这些CAD表示时，其控制点（即图2中的红点）可直接用作形状优化中的设计变量。然而，采用CAD的几何参数化会限制设计过程，因为给定的样条可能无法表示与初始设计差异显著的形状。因此，如果要避免修改样条参数化（例如插入额外的控制点线），这一限制就会使CAD样条仅适用于形状更新幅度较小的情况，例如压铸或模具设计 [12]。

边界 B样条的另一种替代方法是自由形变 [20]。在自由形变中，首先在待变形物体周围构建一条样条（通常是体样条）。其次，对该体样条进行变形，最后将所得变形场施加到其包围的物体上。图3展示了这一过程。自由形变的优势在于，该样条

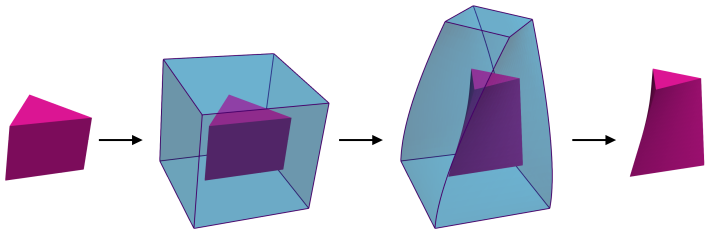


图3：使用体样条（浅蓝色）对混合元件（粉色）进行自由形变。图中省略了控制点。嵌入其中的形状会随简单的体样条嵌入结构相应变形。

的构建独立于其包围的形状，因此在选择次数和分辨率时拥有完全的自由度。这种自由度使设计者能够根据自身需求调整样条（而不必使用针对 CAD 应用优化的既定参数化方式）。因此，自由形变得到了广泛应用，近期 Lassila 和 Rozza 将自由形变与降阶建模相结合的工作便是一例 [36]。我们将把这两种方法——边界 B样条和自由形变——的结合，与一种新颖的、基于神经网络的形状参数化方法进行比较。该方法使用自由形变作为通用接口来修改任意给定的 CAD 样条，进而更新仿真域的边界 [16]。

3. 使用神经网络进行形状参数化

如第2节所述，本研究的首要目标是探究如何利用神经网络，将不同形状编码为一组由少量连续变量组成的表示。为了训练网络并确定这种精简表示，必须向其提供合适的数据。这里所说的合适，是指输入数据（即形状）的提供方式能够让网络从中学习。此外，我们还需要能够使用相同的数据格式，根据神经网络的输出生成高质量的计算网格。

下文将先介绍深度生成模型，再描述一种满足上述两项要求的形状表示。最后，我们将讨论训练数据的生成，以及如何将神经网络用作形状生成器。

3.1. 深度生成模型

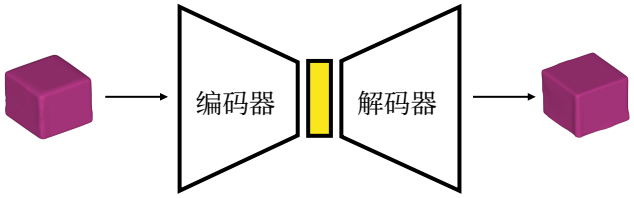
随着生成模型的出现，形状参数化有了另一种实现途径。本小节将回顾两种最常见的生成模型方法，介绍其基本概念和用途，并详细说明如何将它们用于几何过滤。

生成模型是神经网络的一种应用，因此本质上属于分类算法。分类在这里指的是判断某个对象在一定程度上是否接近指定的输入。与单纯对输入进行分类不同，这类模型还可以用来生成与输入相似的输出。不过，「相似」需要进一步解释。在大多数应用中，用户并不希望完全复现给定的输入，而是希望输出类似于输入（即输出应略有变化）。生成

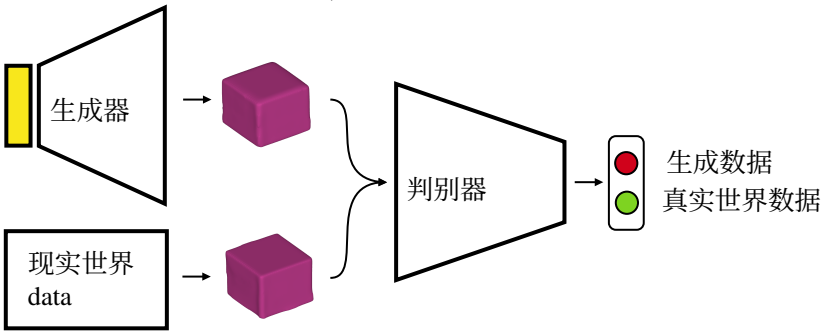
模型试图通过统计建模来实现这一目标。 [37],中有一篇关于生成模型的优秀指南，尤其重点介绍了变分自编码器（VAE）。

VAE 与传统自编码器一样，由编码器 和解码器 组成，旨在 reproducing 给定的任意数据，同时使输入经过一个瓶颈。然而，其通过所谓的“重参数化技巧”实现的概率表述，在实践中相较于传统自编码器具有显著优势 [38]。编码器和解码器的作用可以理解为两个独立的过程。编码器学习给定数据中的关系，并将其编码为所谓的潜变量 **z**。给定这些潜变量后，解码器则学习生成可能 与输入相匹配的数据。训练完成后，用户可以省略编码器，直接通过从潜在空间中采样来生成新数据。详情请参阅 [37, 38], ，应用案例请参阅 [24, 39] 和[40]。

基于样条的方法与生成模型之间的区别在于潜变量的选择。当人工设计者创建一种能够以期望方式修改几何形状的样条参数化时，优化变量就是控制点，这些控制点由设计者直观地 放置在 $\mathbb{R}^3$ 中。相比之下，生成模型学习 一个潜在空间，并明确假设单个潜变量不具有直观的解释。因此，数据从高维直观设计空间（在我们的案例中为 $\chi \subset \mathbb{R}^{3\times n}$）压缩到一个难以解释、特征密集的低维潜在空间$Z$中。简言之，生成模型利用神经网络的计算能力，寻找一个密集的分类空间，从中可以进行采样以生成新数据。对于 VAE，该过程如图 4a 所示。



（a）自编码器通过瓶颈（即低维潜在表示）传递数据，同时实现从输入到输出的映射 on.



(b) 生成对抗模型通过推断能够生成与输入中无法区分的输出的表示来学习潜在空间 put.

图4：深度生成网络的两个主要概念：变分自编码器和生成对抗网络 s.

与变分自编码器相竞争的概念是生成对抗网络（生成对抗网络）。其基本结构如图4b所示。生成对抗网络由Goodfellow等人 [41], 首次提出，遵循不同的概念，并训练两个对抗网络，即生成器和判别器。在生成对抗网络中，生成器经过训练以创建模仿真实世界数据的数据，而判别器则试图判断数据集是否为人工生成。在极小极大的机制下，生成器的学习目标是最大限度地提高判别器做出错误判断的概率。

生成对抗网络已被证明是形状建模的优秀工具。例如，Wu 等人将生成对抗网络用于三维形状生成，并展示了其相较于三维变分自编码器的优越性能。他们甚至利用生成对抗网络，根据变分自编码器的输出重建三维模型；该输出用于推断这些图像的潜在表示[42]。正如 [39], Wu 等人还展示了对学习到的潜在表示进行形状插值和形状算术的能力。最近，Ramasinghe 等人 [43] 利用生成对抗网络，通过点云对高分辨率三维形状进行建模。

3.2. 隐式形状表示

神经网络学习低维潜在空间与三维实体之间的映射。为了构建这样的映射，我们首先需要定义如何表示形状（即，定义神经网络实际需要学习的数据）。在介绍本研究所采用的方法之前，我们先回顾标准方法及其局限性。

机器学习中常见的形状表示方式有三种：(1) 体素，(2) 点云，以及 (3) 网格[33]。网格的问题在于，网格拓扑也决定了可能的形状拓扑。相比之下，点云可以表示任意拓扑，但需要指定分辨率。最后，体素既可以表示任意拓扑，分辨率也可变化，但遗憾的是，其内存消耗会随分辨率呈立方级增长。由于这些缺点，本研究采用的网络学习有符号距离函数，其网络配置最初由 Park 等人 [33]提出。

有符号距离函数为每个空间点提供到待编码曲面上最近点的距离。此外，距离的符号还包含该点位于曲面内部还是外部的信息。利用此类连续的有符号距离函数数据，可以将形状提取为其零值等值面，且可采用适合生成网格的任意分辨率。

3.3. 训练集生成

如第 3.1 节所述，训练神经网络需要一组源形状。然而，据作者所知，目前尚无适用于单螺杆挤出机混合元件的形状库。因此，我们介绍一种构建自定义训练集的方法。

为生成合适的训练集，我们首先选定需要考虑的基准形状——在本例中为销钉式和菠萝式混合器。基于这一选择，我们任意推导出共四种基准形状（即三角形、正方形、六边形和圆柱体，参见图 5）——显然，这些形状不足以支持有效且有意义的训练。我们通过自由形变（FFD）的单独应用或组合应用来改变基准形状，以收集足够数量的训练形状。图 5 给出了所应用变形的示例。总共生成了 2659 个训练形状。

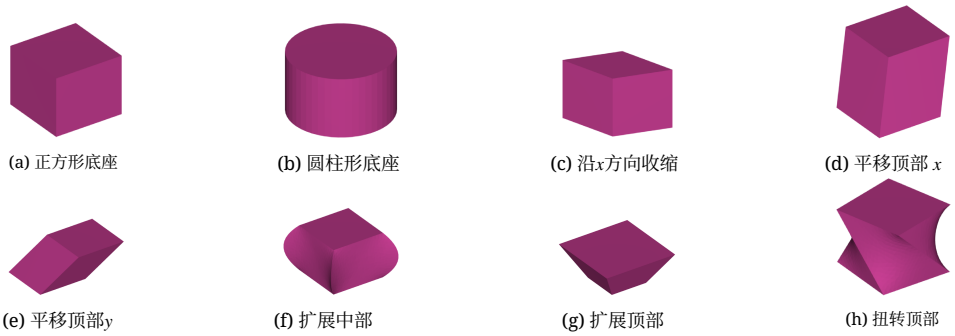


图5：基本形状及其应用形变示例。总共使用了三角形、正方形、圆柱体和六边形作为基准形状。

为从这些形状中获取SDF训练数据，我们采用Park等人 [33] 的方法，首先将每个形状归一化，使其适合置于一个单位球内。然后，我们使用trimesh库 [44]对500,000个空间点及其SDF值对进行采样。

3.4. 形状生成器

如前文所述，形状生成器的任务是根据一组优化变量提供一个混合元件。该形状生成器——在本研究中——因此围绕神经网络构建，下面将对此进行介绍。

所使用的神经网络基于 DeepSDF自动解码器 [33]：这是一个包含十个全连接层的前馈网络，其中八个隐藏层（即内部层）各有 256 个神经元，并采用 ReLU激活函数。与自动编码器不同，自解码器仅训练解码器，并在训练期间同时优化网络参数和潜在编码。我们考察了潜在维度为四、八和十六的情况，l。输入层由这些 l 个神经元与三维查询位置拼接而成。输出层

仅有一个神经元，并采用 tanh 激活函数。关于所选 SDF 网络的详情，我们仍请参阅 [33]。为训练该网络，我们采用 ADAM优化算法 [45]。为使用更优的学习率，我们采用渐进式方法，初始学习率为 $\varepsilon_0 = 5e-4$（对应 $\boldsymbol{\theta}$），以及 $\varepsilon_0 = 1e-3$（对应 $\boldsymbol{z}$），并按如下方式衰减：

$$\varepsilon = \varepsilon_0 \cdot \left(0.5^{e\%500}\right), \tag{3}$$

其中 e 表示当前训练迭代（即 训练轮次），% 表示整数除法。网络的训练可以看作是对形状进行参数化。

为从训练后的网络的 SDF 输出中提取等值面（即生成新的混合元件），我们对离散 SDF 场进行采样，并在 [47] 的实现中应用移动立方体算法 [46]。最后，我们使用 TetGen [48] 进行自动网格划分，获得如图 6 所示的仿真域，其中包含新混合元件。

4. 所开发的形状优化框架

一般而言，我们的框架由三个构成模块组成：(1) 形状生成器、(2) 流动求解器和 (3) 优化器，下文将对其进行介绍。

从一组初始优化变量 $\boldsymbol{\sigma}_0$ 开始，形状生成器会创建一个新的混合元件 $\Omega(\boldsymbol{\sigma}_0)$。随后，流动求解器计算该混合元件周围的流场，优化器则对其进行评估，以确定流体的混合程度。根据得到的混合值，并与之前的迭代结果进行比较，优化算法会确定一组新的优化变量。该过程会反复执行，直至达到最大迭代次数，或满足其他终止准则——通常是获得良好的目标值或目标值下降幅度不明显。

4.1. 流动求解器和仿真模型

流动求解器和仿真模型与 [16] 中介绍的完全相同，因此下面仅作简要概述。通过使用 Carreau 模型和 WLF 温度修正求解稳态、不可压缩、非等温 Navier–Stokes 方程，获得各种混合元件所产生的流场。控制方程采用线性稳定有限元离散，并通过牛顿线性化和 GMRES 迭代求解器求解。随后，我们使用相同的配置求解一组对流方程，以模拟粒子追踪，并将其结果作为目标函数的输入。所有方法均在内部开发的流动求解器中实现。

我们对仿真模型（即单螺杆挤出机流道）作了两项简化：首先，我们仅模拟单个混合元件周围的流动，而非整个混合段。其次，我们假设机筒在展开流道截面中旋转。这两项假设显著降低了计算成本，同时仍能定性评估混合效果的改善情况。为评估混合效果，我们通过求解一系列对流方程来模拟粒子追踪，从而得到熔体流动所携带粒子的入口—出口映射。我们将对流信息处理如下：把入口域的一部分细分为较小的矩形子域。在每个矩形中，我们选取一组粒子，使粒子集合的包围盒与该矩形子域重合。接着，追踪每个粒子通过整个区域的过程，记录其在出口域的位置，最后在出口处对相同的点集构建凸包。对入口与出口处凸包的长度增量取平均，可得到一个简单而稳健的目标函数，其设计灵感源于界面面积测量。利用这一目标函数，我们发现，该仿真模型在准确性与计算效率之间实现了良好平衡 [16]。图 6 展示了所选的仿真域。

4.2. 优化器

我们使用开源优化库 Dakota [49] 来推动设计过程。以下选取并介绍两种不同的算法。第一种算法是矩形划分（DIRECT）算法，最早于 [50]中提出。DIRECT 属于分支定界 方法，并使用 n 维三分法迭代划分设计空间。为找到极小值，该算法遵循利普希茨优化方法，通过评估每个分区中目标值的下界，确定应进一步采样的设计空间分区。选择下界最低的分区并对其进一步采样。DIRECT 对这一

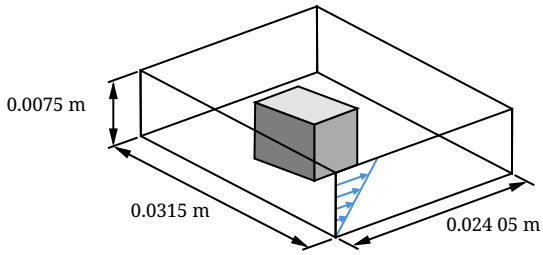


图 6：包含单个混合元件的仿真域，其形状与展开螺杆通道中单个混合元件周围的流动相似。流动条件以蓝色显示，并采用机筒旋转设置。关于目标函数和控制方程的详细说明，请参阅 [16]。

概念进行了改进，计算多个下界，并根据当前采样值（即分区中心的目标值）赋予其权重。这样可以进一步采样相对于分区大小而言目标值较优的分区，从而有效采样尚未探索的大片设计空间。因此，DIRECT 能够识别多个可能最优 分区，并实现全局收敛。

本文采用的第二种算法是单目标遗传算法（SOGA），其多目标变体最初在 JEGA 软件包中提出 [51]。作为遗传算法的一种，它通过模拟生物进化来求解优化问题。因此，每次优化运行都包含大量样本，这些样本称为种群。种群中的个体两两配对并以特定方式重组，使适应度（即目标值）逐步提升。就其在本文中的应用而言，尤其值得注意的是，模拟进化的过程包含一个变异步骤，该步骤会随机修改或重新初始化设计变量。引入的随机性使算法能够跳出设计空间中的局部凸区域。此类进化优化方法通常收敛较慢，导致计算成本较高。不过，它们往往能够找到比非进化算法更好的结果。对于 DIRECT 和 SOGA，我们均采用默认的收敛准则，并将最多 1000 次迭代作为终止准则。完整的计算框架如图 7 所示。

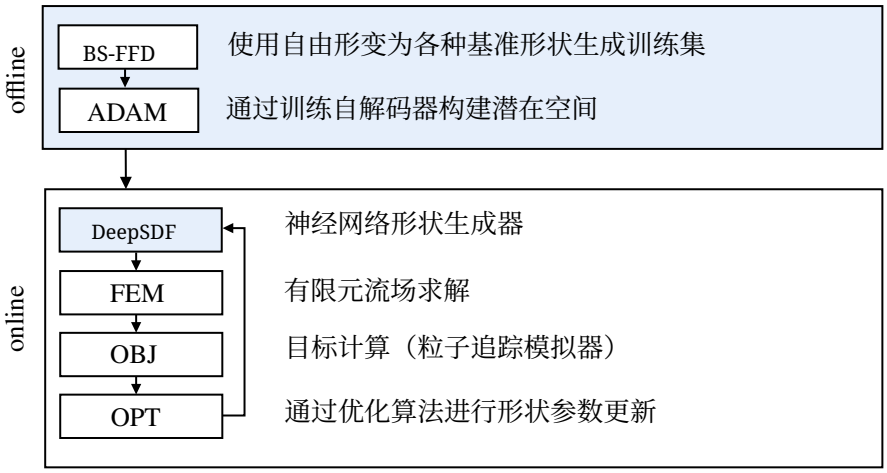


图7：所提出计算框架的构建模块流程。该流程分为两部分：一次性、计算量较大的训练阶段，以及实际优化阶段，其中包括快速筛选评估。为创建训练集，对一组基准形状应用自由形变。随后，我们使用 ADAM优化器训练网络，至此离线阶段结束。在优化过程中（即在线阶段），首先由神经网络生成一个新形状。接着，在该形状周围创建新的计算网格，并基于有限元模拟评估新设计的混合情况。根据目标值，使用经过调整的潜变量重新启动优化循环。与通用的基于几何内核的方法（参见图1）相比有所修改的构建模块以蓝色突出显示。

5. 数值结果

本节介绍采用神经网络形状参数化方法获得的结果。

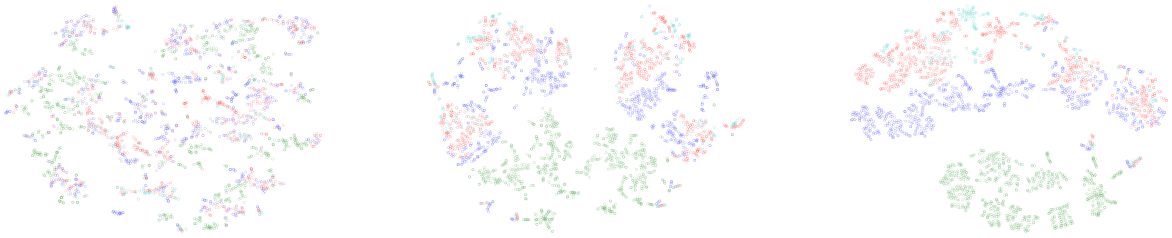
因此，第 5.1 节重点介绍离线阶段的结果，即形状表征神经网络的训练情况。具体而言，我们将讨论所构建的潜在空间如何随其维度而变化，并使用广泛应用的数据降维技术 *t*-分布随机邻域嵌入 *(t-SNE)*来可视化学习得到的 n 维形状参数化。随后，我们将在第 5.2 节介绍通过我们的形状优化方法得到的混合形状。

5.1. 潜在空间维度

最重要的选择之一是嵌入空间的目标维度 *l*。在径向基函数、自由形变、基于 CAD 的方法，乃至基于网格的方法等各种成熟的过滤机制中，实践者都必须在提高灵活性与计算开销之间取得平衡。尽管神经网络有望实现更紧凑、稠密的嵌入，但这一问题仍然重要，并体现在所选潜在空间的维度上。以往研究仅使用了很少的优化变量。Elgeti 等人仅采用一个或两个参数 [12]。然而，作者的其他研究表明，即使采用六个设计变量，也能获得良好的结果 [16]。为了将优化变量的数量控制在较小水平，我们研究了维度分别为四、八和十六的嵌入空间，并与使用九个变量的自由形变方法进行比较。

尽管如第 3.1 节所述，通常获得的潜在空间缺乏直观的解释方式，但我们仍希望评估所学习嵌入空间的质量。我们通过以下三种不同方式进行评估：(1) 展示一种降维技术，以便直观地考察潜在空间；(2) 对两个训练形状的潜在表示进行插值，并将结果与预期结果进行比较；(3) 运用形状算术，即从一个基准形状中分离出特定修改，并将其施加到另一个基准形状上，以检验潜在空间是否也能识别这些特征。

(1) 为了可视化高维潜在空间，需要采用降维技术。一个直观的选择可能是主成分分析（PCA），但 PCA 主要致力于保留全局结构，因此，在高维数据中相距较远的数据点，在二维图中也会相距较远。相反，相似点之间的相关性往往会丢失。由于我们的目标是考察从人的角度来看，相似形状是否由相似的潜在编码表示，因此相似数据相关性的丢失会造成问题。不过，t-SNE [52] 可以缓解局部相关性的丢失问题。使用 t-SNE，我们绘制每个训练形状所获得的潜在编码；由于局部相似性得以保留，相似的潜在编码会在散点图中形成簇。随后可以从这些簇中抽取样本，以验证潜在编码簇是否对应于相似形状。图 8 展示了全部三种潜在维度——4、8 和 16——对应的 t-SNE 图。图 8 展示了



(a) 训练集以4维潜在编码表示　(b) 训练集以8维潜在编码表示　(c) 训练集以16维潜在编码表示

图8：使用不同潜在空间维度得到的t-SNE图。潜在维度增加似乎能提升神经网络的分类性能。每种颜色对应一种基础训练形状：绿色表示三角形基体，深蓝色表示立方体，红色表示六面体，而浅蓝色表示圆柱体的镶嵌版本。

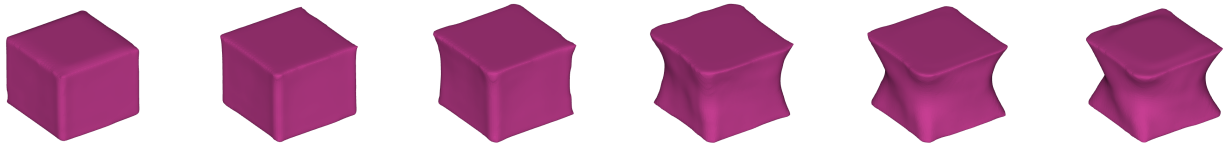
潜在维度的增加会提升神经网络的分类性能。具体而言，随着潜在维度

增加，选取的四种基准形状及其各自的变体聚集得越来越密集。这种分类性能的提升表明，在八维和十六维的情况下，神经网络能够恰当地学习相似形状之间的相似性。

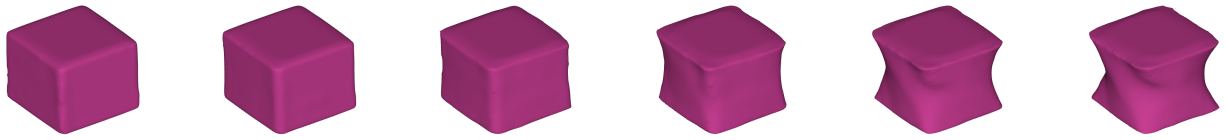
(2) 除了比较物理空间和潜在空间中相似形状的簇之外，我们还研究了潜在空间表示训练集中未包含形状的能力。为此，我们对两种形状进行插值。图9展示了三个潜在空间各自得到的结果。与观察到的不足相一致



(a) 四维形状插值显示重构形状中存在伪影，即潜在表示质量较差。



(b) 八维形状插值的结果令人满意。



(c) 十六维形状插值，与八维潜在表示相比，形状表示仅略有改善。

图9：使用不同潜在维度进行形状插值。插值形状通过$z^{interp} = z_a + z_b - z_{aN+1}\, n$ 得到，其中z_a 和z_b 表示形状a （此处为未变形立方体）与b （此处为扭曲立方体）之间的潜在编码。当$N = 20$时，所示示例表示$n \in [1, 3, 7, 13, 17, 20]$。

在四维潜在空间的分类能力方面，图 9a 显示，形状之间的插值会产生不理想的结果。具体而言，可以观察到形状缺陷。这可能是因为从最右侧的图中可以看出，扭曲立方体在潜在空间中的表示非常不充分。然而，八维和十六维潜在空间都显示出正立方体形状与扭曲立方体形状之间视觉上平滑的过渡。

（3）上述两项分析考察了神经网络的整体分类能力，以及其表示中间形状的适用性。最后一项测试通过应用 形状算术 进行。利用对潜在编码执行的算术运算，我们通过对拉伸立方体和正立方体进行逐分量差值，提取出一个示例特征——此处为沿中心平面的拉伸。该差值表示中心平面扩张，随后可以将其应用于任何其他基准形状——此处为未变形的六面体。图 10 展示了所得形状。同样，由于基准形状未得到细致表示，四维潜在空间的表现明显较差。与插值情况相反，十六维潜在空间现在显示出比八维潜在空间更好的结果。

三项研究——t-SNE图、插值和算术运算——均表明，四维潜在空间无法生成合适的潜在表示。不过需要指出的是，考虑到优化变量数量翻倍，与使用八个潜变量相比，使用十六个潜变量所能带来的收益似乎小得不太吸引人。

5.2. 优化结果

为研究这种新型形状参数化技术的效果，我们比较了潜在空间维度和优化算法各不相同的配置，如表 1 所示。此外，我们要求所有生成的形状体积都与基于样条的优化中使用的未变形菱形混合元完全相同（参见第 4.1 节）。我们采用这种缩放方式，以避免优化收敛到仅仅尺寸增大的形状；这类形状虽能获得较好的目标



(a) 4维形状运算存在明显的表示误差，尤其是六面体和最终形状。



(b) 8维形状运算的表示效果优于4维潜在编码，但结果仍略有偏差。



(c) 16维形状运算与所有训练形状都完美相似，且生成的形状也很规整。

图10：不同潜在维度下的形状运算。通过将潜在编码计算为 $z^{E4_{thick}} - z_{E4} + z_{E6}$，在六边形基体的中心平面上施加线性加厚，其中 $z^{E4_{thick}}$、z_{E4} 和 z_{E6} 分别表示加厚立方体、正立方体和正六面体的潜在编码。

值，却无法提供有益的见解。表 1 列出了所得结果，表 2 则展示了相应的计算工作量。获得的最佳形状如图 11 所示。对比优化几何形状

	4	8	16	FFD
SOGA	-0.0726	-0.0710	-0.0750	–
DIRECT	-0.0645	-0.0738	-0.0769	-0.0422

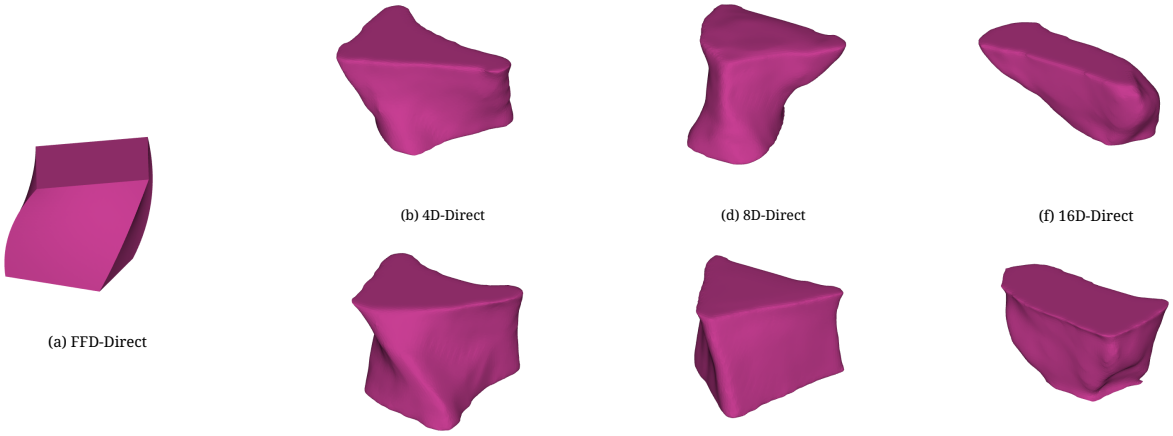
表1：比较不同优化算法和潜在空间维度的最佳目标值，并与九维自由形变进行对照。使用上述目标函数形式1b时，值越小表示结果越好。

	4		8		16		FFD	
# 迭代次数	最优	总计	最优	总计	最优	总计	最优	总计
SOGA	768	1000	752	1000	534	1000	–	–
DIRECT	96	113	129	143	138	149	16	67

表 2：比较不同优化算法和潜在空间维度的最终迭代次数、所得目标值及总迭代次数；并与九维自由形变进行对比。

从塑料加工的角度来看，这些结果颇有启发。一方面，三角形形状以及顶部逐渐变宽的混合元件似乎更具优势。不过需要注意的是，这些形变并不代表塑料工程中的普遍最优解，而只是在训练集所允许的范围内可能达到的最佳形变。预计选用更多样化的训练集还能得到进一步改进的形状。

对于本研究而言（重点关注将神经网络用作形状参数化），更相关的是比较收敛情况、达到的混合效果，以及结果之间的差异与相似之处。表1显示，对于所选的形状优化问题，DIRECT算法与SOGA相比并无劣势，且能够可靠地收敛。同时，形状参数化的维度似乎会影响优化结果，因为四维和八维神经网络得到的是优化三角形。相比之下，十六维的情况则使



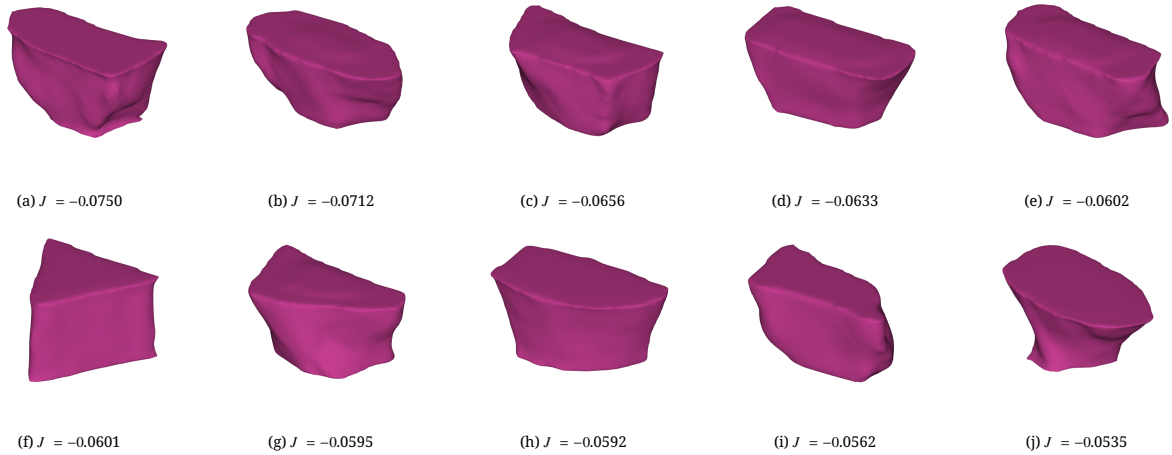
(a) FFD-Direct (b) 4D-Direct (d) 8D-Direct (f) 16D-Direct

(c) 4D-Soga (e) 8D-Soga (g) 16D-Soga

图11：优 针对所有不同潜在编码和优化算法获得的优化结果，与现有的基于FFD的形状优化进行比较。

顶部扩张的四边形成为最优形状。所有结果的共同点是几何形状均有倾斜，并略微扭曲。

基于样条的形状优化与基于神经网络的形状优化之间一个显著的区别是，基于神经网络的形状参数化能够编码多个形状，其中不止一种都可能同样擅长混合熔体。因此，从实践者的角度来看，不应只关注最佳结果，而应比较多个同样最优的设计，并从中归纳设计规则，这确实是有意义的。图 12 展示了这样的比较，并揭示了进化算法的一项优势。DIRECT算法会收敛到局部解，因此排名前十的设计在几何形状上相似；而 SOGA 的生成特性则使实践者能够找出可能同样有效的设计（参见图 12f 和 g），并从中选择最经济的方案。这样便能兼顾螺杆清洁、可制造性及其他方面的限制。



(a) $J = -0.0750$ (b) $J = -0.0712$ (c) $J = -0.0656$ (d) $J = -0.0633$ (e) $J = -0.0602$

(f) $J = -0.0601$ (g) $J = -0.0595$ (h) $J = -0.0592$ (i) $J = -0.0562$ (j) $J = -0.0535$

图12：通过16维SOGA优化获得的十种最佳形状。除排名第六的形状（h）外，所有形状的顶部均有所扩展，方向相似，并且在y方向上显得更宽（即垂直于主流动方向）。

6. 讨论与展望

在本研究中，我们探讨了生成模型作为形状参数化方法的适用性，并以动态混合元件的数值形状优化作为应用案例。所开发的形状参数化方法的基本原理，是利用神经网络构建从高维空间到特征丰富的低维潜在空间的降维映射。

首先，通过 t-SNE 图研究这一低维空间的性质。这些图提供了直观证据，表明生成模型能够构建平滑的形状参数化，从而可以采用经典的启发式优化算法。将遗传算法与此类启发式算法进行比较，表 2 显示，SOGA 算法所需的迭代次数（即仿真次数）显著更多。此外，表 1 显示，在所研究的示例中，额外增加的计算工作量并未按比例转化为混合效果的提升。人们可能会认为，SOGA 算法的随机性或许更适合探索难以解释的潜在空间。然而，结果表明，所学习到的参数化具有平滑性，因此像 DIRECT 这样的确定性方法同样适合在潜在空间中进行优化。

除了研究生成模型的普遍适用性外，我们还考察了不同潜在维度的影响。虽然实际优化结果看起来令人满意，但图 9 和图 10 表明，过度压缩（即四维）的潜在空间可能不适用于优化。类似地，从优化结果中无法直接判断八维和十六维哪种更优。同样，图 10 表明，更高维度的潜在空间能够实现更精确的形状编码，这通常更可取。由于优化问题收敛前的总迭代次数相近，因此，与八维方案相比，可以选择十六维参数化。

如预期所示，相较于成熟的低维形状参数化方法，这种新方法的一项根本性改进在于，它能通过一次优化覆盖广阔得多的设计空间。由于其核心理念是对多种形状进行编码，优化会产生许多几乎同样优秀的形状。因此，这种新方法对现有方法有所拓展，使实践者能够推导出 最能提升混合效果、且适用于多种基准形状的设计特征。因此，迈向最优设计的关键步骤并非构建复杂的形状参数化，而是创造性地定义训练集。

最后，使用基于神经网络的形状参数化时，一个重大挑战是妥善控制输出形状的尺寸。本研究采用体积约束，以避免混合元件仅仅朝着尺寸最大化的方向发展。不过，重新构造目标函数（例如对压力损失施加惩罚）或许能够避免这类不利设计。或者，也可以将缩放因子作为额外的优化变量。尺寸控制和高效的训练集生成都可能成为后续研究的课题。

鉴于上述结果，利用信息丰富的潜在表示及其强大的泛化能力，有望显著改进现有的工业设计。

7. 致谢

谨此致谢德国研究联合会（DFG）通过DFG资助项目“单螺杆挤出机动态混合与剪切元件的自动化设计和优化”及优先计划2231“高效冷却、润滑和输送——面向高效生产过程的机械与流体动力学耦合仿真方法（FLUSIMPRO）”提供的资助，项目编号439919057。相关实施工作在亚琛工业大学IT中心提供的高性能计算集群上完成。仿真使用了亚琛工业大学在项目 jara0185 和 thes0735 下提供的计算资源。

参考文献

[1] O. Celik, T. Erb, C. Bonten，预测单螺杆挤出机中的混合质量，《塑料》71 (2017) 175–177。 [2] M. Gale，《单螺杆挤出中的混合》，手册系列，Smithers Rapra出版社，2009。 [3] W. Roland, C. Marschik, J. Miethlinger，C. Chung，不同菠萝式混合器设计的混合研究——仿真结果 1，2019。 [4] X. Sun, M. A. Spalding, T. W. Womer, N. Uzelac，采用数值模拟优化单螺杆挤出用麦多克混合器设计，载于：塑料工程师协会第75届年度技术会议论文集（ANTEC），2017。 [5] G. A. Campbell, M. A. Spalding，《单螺杆挤出机的分析与故障排除》，载于：《单螺杆挤出机的分析与故障排除》（第二版），第二版，Hanser，2021，第 I–XIX 页。 doi:https://doi.org/10.3139/9781569907856.fm。网址：https://www.sciencedirect.com/science/article/pii/B9781569907849500011 [6] C. Marschik, T. Osswald, W. Roland, H. Albrecht, O. Skrabala, J. Miethlinger，Block-Head混合螺杆中的混合数值分析，《聚合物工程与科学》59 (11 2018)。 doi:10.1002/pen.24968。 [7] 非牛顿流体流体力学，第一版，《应用数学与力学指南》——Teubner Studienbücher，Vieweg+Teubner出版社，威斯巴登，1981。 [8] S. J. Kim, T. H. Kwon，通过混沌流增强单螺杆挤出过程的混合性能：第一部分，基本概念与实验研究，《聚合物技术进展》15 (1) (1996) 41–54。doi:https://doi.org/10.1002/(SICI) 1098-2329(199621)15:1<41::AID-ADV4>3.0.CO;2-K。 [9] A. C.-Y. Wong, Y. Lam, A. C. M. Wong，通过可视化和图像分析量化不同配置单螺杆的动态混合性能，《聚合物技术进展》28 (1) (2009) 1–15。 doi:https://doi.org/10.1002/adv.20142。 [10] S. J. Kim, T. H. Kwon，通过混沌流增强单螺杆挤出过程的混合性能：第二部分，数值研究，《聚合物技术进展》15 (1) (1996) 55–69。doi:https://doi.org/10.1002/(SICI)1098-2329(199621)15:1<55:: AID-ADV5>3.0.CO;2-J。 [11] N. Domingues, A. Gaspar-Cunha, J. Covas，一种评估聚合物挤出用单螺杆挤出机混合能力的定量方法，《聚合物工程杂志》32 (03 2012)。 doi:10.1515/polyeng-2012-0501。 [12] S. Elgeti, M. Probst, C. Windeck, M. Behr, W. Michaeli, C. Hopmann，作为挤出模具设计方法的数值形状优化，《有限元分析与设计》61 (2012) 35–43。doi:10.1016/j.finel.2012.06.008。 [13] R. Siegbert, M. Behr, S. Elgeti，将口模膨胀作为聚合物挤出模具设计的目标，《AIP会议论文集》1769 (1) (2016) 140003。doi:10.1063/1.4963540。 [14] A. Gaspar-Cunha, J. Covas，挤出螺杆设计：一种优化方法，《国际聚合物加工》16 (09 2001)。doi:10.3139/217.1652。 [15] H. Potente, W. H. T¨obben，基于三维有限元模拟和新计算模型改进剪切段设计，《大分子材料与工程》287 (11) (2002) 808–814。 doi:https://doi.org/10.1002/mame.200290010。 [16] S. Hube, M. Behr, S. Elgeti, M. Sch¨on, J. Sasse, C. Hopmann，分布混合元件的数值设计，《有限元分析与设计》204 (2022) 103733。doi:https://doi.org/10.1016/j.finel.2022.103733。网址：https://www.sciencedirect.com/science/article/pii/S0168874 22000117 X M. Botsch, L. Kobbelt，实时自由形态建模的直观框架，《ACM图形学汇刊》23 (3) (2004) 630–634。doi:10.1145/ 1015706.1015772。 [18] G. Farin，5——贝塞尔曲线的伯恩斯坦形式，载于：G. Farin（编），《计算机辅助几何设计中的曲线与曲面》（第五版），第五版，Morgan Kaufmann计算机图形学丛书，Morgan Kaufmann，旧金山，2002，第57–79页。doi:https://doi.org/10.1016/ B978-155860737-8/50005-3。网址：https://www.sciencedirect.com/science/article/pii/B9781558607378500053 [19] L. Piegl, W. Tiller，《NURBS手册》，视觉传达专著丛书，施普林格·柏林·海德堡，1996。 [20] T. W. Sederberg, S. R. Parry，实体几何模型的自由形变，《SIGGRAPH计算机图形学》20 (4) (1986) 151–160。doi:10.1145/15886.15903。 [21] M. Hojjat, E. Stavropoulou, K.-U. Bletzinger，用于基于节点的形状优化的顶点变形方法，《计算力学与工程中的计算方法》268 (2014) 494–513。doi:https://doi.org/10.1016/j.cma.2013.10.015。网址：https://www.sciencedirect.com/science/article/pii/S0045782513002673 [22] J. Johnson, A. Alahi, F. Li，用于实时风格迁移和超分辨率的感知损失，CoRR abs/1603.08155 (2016)。arXiv: 1603.08155。

[23] T. Friedrich, N. Aulig, S. Menzel，关于神经风格迁移应用于三维形状数据的潜力与挑战，载于：EngOpt 2018，施普林格，计算机科学讲义，人工智能讲义，生物信息学讲义，2018。[24] S. Liu, A. G. O. II, C. L. Giles，学习体素化三维形状的层次潜变量模型，CoRR abs/1705.05994 (2017)。arXiv: 1705.05994。[25] J. Wu, C. Zhang, T. Xue, W. T. Freeman, J. B. Tenenbaum，通过三维生成对抗建模学习物体形状的概率潜在空间，载于：神经信息处理系统进展，2016，第82–90页。[26] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, J. Xiao，三维ShapeNet：体积形状的深度表征（2014）。arXiv: 1406.5670。[27] C. R. Qi, H. Su, K. Mo, L. J. Guibas，PointNet：用于三维分类和分割的点集深度学习（2016）。arXiv:1612.00593。[28] S. Ramasinghe, S. Khan, N. Barnes, S. Gould，用于高分辨率三维点云生成的谱生成对抗网络（2019）。arXiv:1912.01800。[29] Y. Yang, C. Feng, Y. Shen, D. Tian，FoldingNet：通过深度网格变形实现的点云自编码器（2017）。arXiv:1712.07262。[30] T. Bagautdinov, C. Wu, J. Saragih, P. Fua, Y. Sheikh，使用组合变分自编码器建模面部几何，载于：2018年IEEE/CVF计算机视觉与模式识别会议，2018，第3877–3886页。[31] Q. Tan, L. Gao, Y.-K. Lai, S. Xia，用于变形三维网格模型的变分自编码器，2018，第5841–5850页。doi:10.1109/CVPR. 2018.00612。[32] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, M. Aubry，AtlasNet：一种学习三维曲面生成的纸浆塑形方法（2018）。arXiv:1802.05384。[33] J. J. Park, P. Florence, J. Straub, R. Newcombe, S. Lovegrove，DeepSDF：学习用于形状表示的连续有符号距离函数，载于：IEEE计算机视觉与模式识别会议（CVPR），2019。[34] Z. Chen, H. Zhang，学习用于生成式形状建模的隐式场，IEEE计算机视觉与模式识别会议论文集（CVPR）（2019）。[35] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, A. Geiger，占用网络：在函数空间中学习三维重建，载于：IEEE计算机视觉与模式识别会议论文集（CVPR），2019。[36] T. Lassila, G. Rozza，基于偏微分方程模型和降阶基方法的参数化自由形状设计，《计算力学与工程中的计算方法》199 (23) (2010) 1583–1592。doi:https://doi.org/10.1016/j.cma.2010.01.007。[37] C. Doersch，变分自编码器教程（2021）。arXiv:1606.05908。[38] D. P. Kingma, M. Welling，自编码变分贝叶斯（2014）。arXiv:1312.6114。[39] S. Liu, C. L. Giles, A. G. O. I. au2，学习三维形状的层次潜变量模型（2018）。arXiv:1705.05994。[40] Q. Tan, L. Gao, Y. Lai, S. Xia，用于变形三维网格模型的变分自编码器，载于：2018年IEEE/CVF计算机视觉与模式识别会议，2018，第5841–5850页。doi:10.1109/CVPR.2018.00612。[41] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio，生成对抗网络，载于：Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, K. Q. Weinberger（编），《神经信息处理系统进展》第27卷，Curran Associates, Inc.，2014，第2672–2680页。[42] J. Wu, C. Zhang, T. Xue, B. Freeman, J. Tenenbaum，通过三维生成对抗建模学习物体形状的概率潜在空间，载于：D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett（编），《神经信息处理系统进展》第29卷，Curran Associates, Inc.，2016，第82–90页。[43] S. Ramasinghe, S. Khan, N. Barnes, S. Gould，用于高分辨率三维点云生成的谱生成对抗网络（2020）。arXiv:1912.01800。[44] Dawson-Haggerty等，trimesh。网址：https://trimsh.org/ [45] D. P. Kingma, J. Ba，ADAM：一种随机优化方法（2017）。arXiv:1412.6980。[46] W. E. Lorensen, H. E. Cline，行进立方体：一种高分辨率三维表面构造算法，《SIGGRAPH计算机图形学》21 (4) (1987) 163–169。doi:10.1145/37402.37422。[47] T. Lewiner, H. Lopes, A. W. Vieira, G. Tavares，具有拓扑保证的行进立方体案例高效实现，《图形工具期刊》8 (2003) 2003。[48] H. Si，TetGen：基于德劳内三角剖分的高质量四面体网格生成器，《ACM数学软件汇刊》41 (2) (2015年2月)。doi:10.1145/2629697。网址：https://doi.org/10.1145/2629697。/ B. Adams, M. Eldred, G. Geraci, R. Hooper, J. J.D., K. Maupin, J. Monschke, A. Rushdi, J. Stephens, L. Swiler, T. Wildey, Dakota：用于设计优化、参数估计、不确定性量化和敏感性分析的多层并行面向对象框架：6.10版用户手册（2014年7月；2019年5月更新）。[50] D. Jones, C. Perttunen, B. Stuckman，不使用利普希茨常数的利普希茨优化，《优化理论与应用杂志》79 (1993) 157–181。doi:10.1007/BF00941892。[51] J. Eddy, K. Lewis，利用遗传编程有效生成帕累托集，第2B卷：第27届设计自动化国际设计工程技术会议暨计算机与信息工程会议，2001，第783–791页。arXiv:https://asmedigitalcollection.asme.org/IDETC-CIE/proceedings-pdf/IDETC-CIE2001/80234/783/ 6590957/783_1.pdf, doi:10.1115/DETC2001/DAC-21094。网址：https://doi.org/10.1115/DETC2001/DAC-21094。/DETC / DAC- L. van der Maaten, G. Hinton，使用t-SNE可视化高维数据，《机器学习研究期刊》9 (11) (2008) 2579–2605。