关于形状质量多样性优化中表示的适用性

预印本

Ludovico Scarton*波恩-莱茵-锡格应用技术
大学计算机科学系 圣奥古斯丁，53757，德国
ludovico.scarton@smail.inf.h-brs.de

Alexander Hagg技术、资源与能源高效工程研究所（TREE）波恩-莱茵-锡格应用科学大学 圣奥古斯丁，53757，德国
alexander.hagg@h-brs.de

2023年4月6日

arXiv:2304.03520v1 [cs.NE] 7 Apr 2023

摘要

进化算法中使用的表示或编码会对其性能产生重大影响。对机器人领域中广泛使用的表示是否适用于质量多样性优化（QD）的研究，对于最合适的编码方法得出了不一致的结果。鉴于 QD 具有领域依赖性，还需要来自其他领域的证据。本研究比较了多种表示对建筑领域中生成体素网格的影响，包括直接编码、基于字典的表示、参数化编码、组合模式生成网络和元胞自动机。结果表明，某些间接编码优于直接编码，并能生成更多样化的解集，尤其是在考虑完整的表型多样性时。本文提出了一种多编码 QD 方法，将所有评估过的表示纳入同一个档案。不同编码种类依据表型特征展开竞争，由此形成的方法表现与最佳单编码 QD 方法相近。值得注意的是，这种方法并不总是需要最佳单编码的贡献。

关键词 编码、表示、质量多样性、组合模式生成网络、元胞自动机、参数化

1 引言

质量多样性优化是一种进化范式，其基础是将搜索空间、基因组和利基空间分离开来，利基空间由描述解（即表型）特定方面的特征组成。该范式中的算法能够创建多样且高性能的解集。通过这种方式，质量多样性可以帮助人们洞察问题的底层结构，为用户提供许多设计变体，并提高解在动态和不确定环境中的稳健性。解的编码（也称为表示）能够创建尽可能高质量且多样化的解集，这一点至关重要。质量多样性优化通常针对具有高维性质的工程解，因为它们包含扩展表型（Hagg, 2021b），例如依赖时间的机器人行走步态策略（Cully et al., 2015）或

*这是最终版本，已被 GECCO 会议接收发表

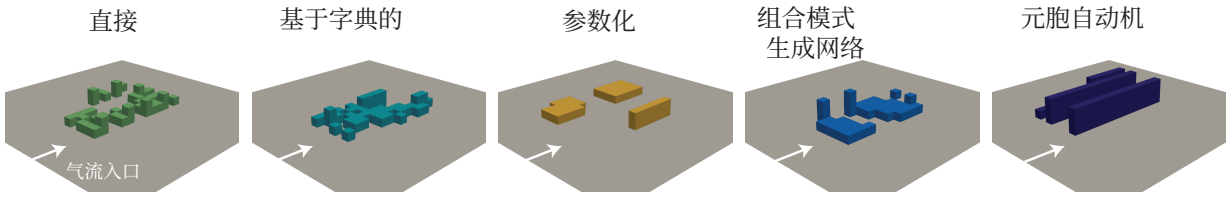

图 1：利用五种表征通过质量多样性优化生成的建筑体量设计示例 s.

建筑物周围气流的形状（Hagg 等，2020）。过去的研究表明，解的编码对进化算法（进化算法）的性能有重大影响，但通常一方面关注进化算法的收敛时间，另一方面关注解的质量，正如（Rothlauf，2006）提出的时间—质量框架所做的那样。一项重要发现是，进化算法的收敛时间往往会随着编码维度的增加而呈超线性增长（Chen等，2015）。但与其他进化算法相比，质量多样性算法高度重视解的多样性。本文旨在深入探讨不同编码不仅对质量，而且对使用质量多样性生成的解集的多样性所产生的影响。

对用于生成机器人神经控制器的各种编码进行比较（Tarapore等，2016）后发现，生成式编码实际上可能会限制质量多样性优化；在他们的研究中，直接编码更容易填满解档案，且生成的解集整体适应度通常高于通过间接编码生成的解。他们的研究证实了这一结论：局部性——即小幅变异会带来表型的小幅变化，而直接编码正具有这一特点——对于质量多样性优化而言，可能比对于更经典的进化算法更为重要。然而，在进化机器人手臂形状时，其他研究发现，间接编码有助于进一步探索设计空间并提高适应度（Collins等，2019）。对于形状领域等质量多样性优化问题，编码的表现仍是一个悬而未决的问题。质量多样性优化更受益于直接编码带来的平滑变异，还是间接编码带来的快速探索？

典型的形状领域是建筑设计，建筑师在最优设计的边界内追求创意自由。在建筑的体量设计阶段，许多设计要求尚未完全确定，而是在此过程中逐步发现的（Maher，2000）。虽然这一阶段允许最大的设计自由，但在此阶段做出的决策会限制后续设计阶段的创意自由和最高设计质量。因此，体量设计阶段是一个非常适合采用质量多样性加以解决的问题，从而在设计过程的早期为城市规划师和建筑师提供参考。建设项目的一项重要质量指标是其气候影响，例如冷气流通量、白天的热吸收和夜间的热辐射，或风环境扰民问题。传统上，这些气候影响会在设计项目的后期阶段进行评估，尤其是通过对具体设计开展流场分析。在这里，质量多样性就可以发挥作用：在设计过程的早期阶段加以应用。

由于建筑设计具有高维性，当编码能将低维基因组映射到这些高维表型时，便能发挥作用。通过限制基因组的维度，间接编码可以缩短搜索过程的收敛时间，因此预计间接编码能够更快地找到高质量的解。同时，这也意味着并非所有表型都能被触及，这可能导致这些编码产生较低的多样性，甚至较低的适应度。优秀的编码只需少量遗传维度，便能找到许多表现优异的解。遗传空间与有用表型集合之间的这种匹配取决于具体领域，这正是本文开展研究的必要性所在。本文聚焦于能够生成体素化三维网格的编码，这种网格是建筑领域中体量设计的一种常见表示。本文分析了多种编码对所得解集多样性和质量的影响，并提供证据以帮助回答以下研究问题：

1. 直接编码是否如 Tarapore等，2016（Tarapore等，2016）所得出的结论一样，比间接编码具有更高的质量多样性性能？ 2. 遗传维度较低是否会带来更高的质量多样性性能，如 Chen 等（2015）（Chen 等，2015）所指出的那样？ 3. 对于当前领域而言，哪种编码最为适用？ 4. 不同编码是否会覆盖表型空间中的不同区域？ 5. 在质量多样性中结合使用多种编码，是否能够提高解集的多样性？

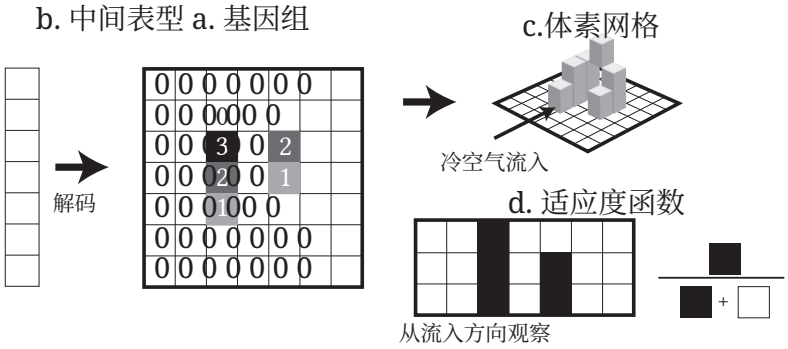

图 2：编码定义了基因组（a）如何解码为表型（b、c）。图示示例包含的网格单元少于本文中实际使用的表型。解的适应度根据冷空气流入方向上已建造单元的相对数量计算（d）。

据我们所知，这是首项系统比较用于质量多样性的体素化三维网格表示的各种广泛使用编码的解集多样性及其他特征的研究。我们相信，我们的发现将引起质量多样性领域研究人员的普遍兴趣，尤其是建成环境或形状设计领域的实践者的兴趣。本工作是针对特定领域的一项初步研究。

2 方法

我们先来定义进化算法（EA）中常用、受生物进化启发的表征术语。EA 中的表征由基因组-表型解码和表型-适应度映射描述（Rothlauf，2006）。表型$\mathbf{p} \in \mathcal{P}$是针对特定问题领域所表达的解。 $\mathcal{X}$定义为 EA 的遗传搜索空间。对于连续优化问题，基因组$\mathbf{x} \in \mathcal{X}$通常由一个实数元组构成。表型 p 通过编码 $\mathcal{E}\colon \mathcal{P} \to \mathcal{X}$编码为基因组$\mathbf{x}$。编码也可与表示一词互换使用。要从基因组中获取 p，需要使用解码过程 $\mathcal{D}\colon \mathcal{X} \to \mathcal{P}$。

根据采用的思路不同，建筑体量设计中的完整表型既可以是设计的形态，也可以是现场设计，例如由此造成建筑周围特定的气流、风环境设计，或夜间特定的热释放。在常见的质量多样性问题领域中，仅用基因组—表型（以及表型—适应度）映射不足以描述表征，因为我们关注的不只是根据环境中的行为得出的适应度指标，也关注行为的其他方面。质量多样性常用于演化机器人控制器的行为策略，或像本研究这样，演化作用于流场的三维体素网格。因此，质量多样性中表征的定义扩展为包括从表型到扩展表型（Dawkins，1982）的映射；扩展表型还包括解在环境中的“行为”（Hagg，2021a）。

本研究将表型定义为由 11×14 个 3×3 米见方单元组成的网格（图 2b），其中包含高度值。扩展表型是在该网格基础上构建的网格，每个单元中包含高度为 0、3、6 或 9 米的体量（图 2c），并置于流场中。问题领域的灵感来自一个受城市规划约束的真实建设项目。表型的维度为 154。城市规划和建筑设计优化中会用到的、计算成本高昂的计算流体力学（CFD），在本研究的实验中由一种非常简单的代理模型取代：面向冷空气流入主方向的相对建成面积（图 2d）。将该面积最小化，可作为降低气流影响的替代指标；这种做法足以产生真实且可解释的结果，同时保持较低的计算需求。体量设计体素网格最直接的表示方式是高度值数组。然而，也存在大量更复杂的表示方式。我们概述了形状优化问题中常用的一些表示方式，这些问题与本文采用的三维体素领域类似。

本节其余部分将介绍 QD算法和若干常见表征，包括它们的变异算子。最后，将介绍一种多重编码 QD算法，它同时使用不同的表征来探索解空间。

2.1 MAP-Elites

最早的 QD算法之一——表型精英多维档案（MAP-Elites）（Cully等，2015）使用类似多维网格的档案作为表型生态位划分空间（见算法1）。其维度由表型特征定义，在本例中分别是总建筑面积和独立建筑物的数量。初始化随机基因组群体 $\mathcal{X}$后，通过解码函数得到其表型 $\mathcal{P}$。根据这些表型，可以推导出个体的质量（适应度）和特征。个体基因组会被放入档案 $\mathcal{A}$中。只有当新个体优于已占据该生态位的个体，或该生态位为空时，才会被接纳。档案作为一个种群，从中随机抽取亲本。亲本 $\mathcal{X}_p$ 经变异算子扰动后，产生子代 $\mathcal{X}_c$。随后，如果原有生态位精英的表现较差，或该生态位为空，子代 $\mathcal{X}_c$ 就会替换档案中属于同一生态位的个体。重复这一过程，直至达到最大代数或满足停止条件。

算法 1 MAP-Elites 算法

$\mathcal{X}, \mathcal{A} \leftarrow Initialize()$ ▷ 初始化基因组和档案
过程 MAP-Elites($\mathcal{X}$, $\mathcal{A}$)
  $\mathcal{P} \leftarrow \mathcal{D}(\mathcal{X})$ 将基因组解码为表型
  $\mathbf{f}, \mathbf{p} \leftarrow Fitness(\mathcal{P})$ ▷ 获取功能与性能
  $\mathcal{A} \leftarrow Replace(\mathcal{A}, \mathcal{X}, \mathbf{f}, \mathbf{p})$ ▷ 替换生态位
  while $gens < maxGens$ **do**
    $\mathcal{X}_p \leftarrow Random(\mathcal{A})$ ▷ 随机选择父代
    $\mathcal{X}_c \leftarrow Perturb(\mathcal{X}_p)$ ▷ 扰动父代
    $\mathcal{P}_c \leftarrow \mathcal{D}(\mathcal{X}_c)$
    $\mathbf{f}, \mathbf{p} \leftarrow Fitness(\mathcal{P}_c)$
    $\mathcal{A} \leftarrow Replace(\mathcal{A}, \mathcal{X}_c, \mathbf{f}, \mathbf{p})$
  结束循环
结束过程

2.2 直接编码

最简单的编码类型将基因组一对一映射到表型（图3）。将基因组转换为表型的解码函数定义如下:

$$\mathcal{D} : \mathbf{p} = \mathbf{x}, \mathbf{x} \in \mathcal{X}, \mathbf{p} \in \mathcal{P} \tag{1}$$

这种编码是最早在进化算法中得到广泛采用的编码方式（Rechenberg，1973；Schwefel 和Schwefel，1977）。对于体素网格问题，体素网格中的每个高度值都将由一个数值表示（通常为实数值，但也可以是二进制值），并在基因组中拥有自己的基因座。基因组的高维度使搜索过程变得繁琐且缓慢，并且预计难以轻易产生高质量的解集。从技术上讲，我们可以预期这种编码能够找到一个非常多样化的解集，因为所有潜在的表型都可以被达到。然而，只有在很长时间内，这才是合理的
由于搜索空间的所有部分都可达，且无需穿过无效区域。搜索过程也可能在这个高维空间中耗费难以估量的时间。

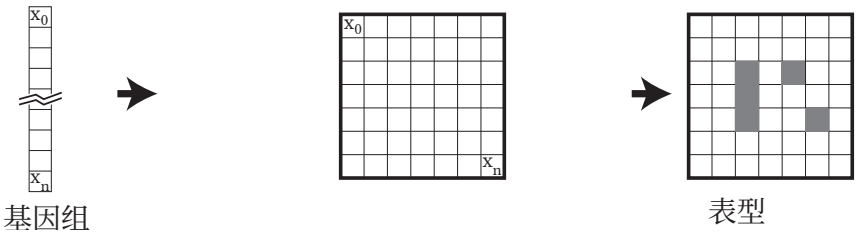

Figure3: 直接编码： 细胞的高度值被直接编码到基因组中 e.

每个网格单元的高度都采用直接编码。因此，这种编码的维度等于表型的维度，即 $d = 154$。解以较小的概率发生变异 p。变异会使某个基因的值增加或减少 1，但其取值仅限于最小高度和最大高度之间。

2.3 基于字典的

为了降低搜索空间的维度，可以使用构建块字典来生成解（图 4）。这种方法常用于平面布局设计优化问题（Modrak 等，2021）。解码函数定义为：

$$\mathcal{D} : \mathbf{p} = \mathbf{DICT}(\mathbf{x}), \mathbf{x} \in \mathbb{N} \tag{2}$$

DICT 是一个构建块字典，根据整数值基因组 **x** 从中选择元素向量 **p** 。该方法曾用于城市布局设计优化（Xu 等，2019），其中表型由单元格组成，其状态由字典中的一个元素决定。搜索空间的维度由字典元素的大小决定。

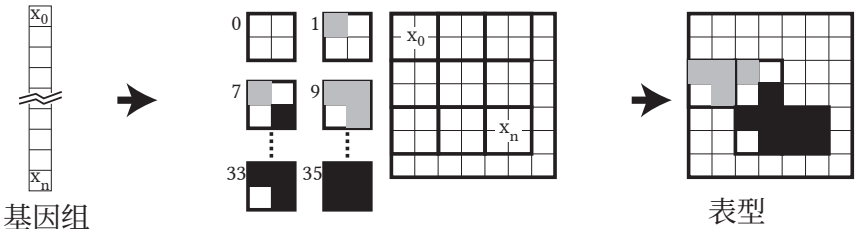

图 4：基于字典的编码：网格单元被分组，并为每个组预先计算所有建筑块。基因组编码了每个单元组被分配到哪个建筑块。

该字典包含大小为 $n \times m$ 的构建块，涵盖所有不会产生悬空体素的体素填充状态。因此，它能够生成与直接编码完全相同的形状。变异算子可用于确定允许哪些状态转换。其定义方式为：以较小的概率 p，按照字典中的允许规则随机执行基因状态转换。变异只会将构建块改变为新旧状态中的单元格曼哈顿距离为 5 的情况，以提供一种更保守的变异算子。

2.4 参数化编码

参数化编码在工程设计中很常见，因为它们易于理解，并且能够提供高度的控制能力。样条编码就是一个例子：通过移动样条的关键点，可以仅调整局部形状特征（Sobieczky，1999）。其他参数化编码还包括一组坐标，用于确定一组立方体的位置或尺寸（Kaushik和Janssen，2013）。作为高度控制能力的代价，这些编码在可生成解的多样性方面往往具有较大的限制。为解决这一问题，研究者开发了更通用的参数化编码，例如加法或减法参数化编码（Wang等人，2019）；这类方法允许用户预先设定设计的全局特征，例如建筑物的数量（Wang等人，2020）。其维度

这类编码的搜索空间通常小于直接编码，但所得解集的多样性可能会受到影响。

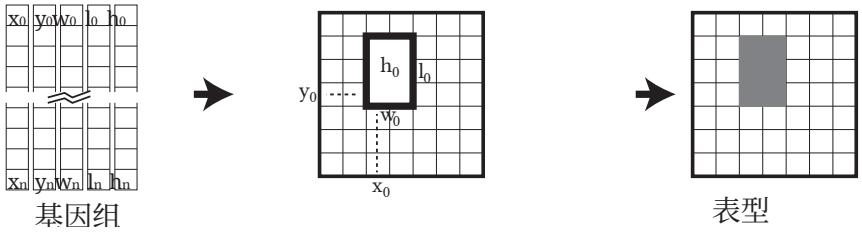

图5：参数化编码： 基因组编码固定数量矩形的位置和长度尺寸。

参数化方式有很多种，但对于本文的问题领域，采用以下定义。解码过程 $\mathcal{D}$ 生成 n 个矩形，由四个参数定义：矩形的 x 和 y 位置，以及长度 l 和宽度 w （图 5）。变异算子以较小的概率 p，根据标准差为σ的上取整正态分布的取值添加一个整数值。矩形不得放置在可建造区域之外。确保 w 和 l 均为非负值。

2.5 组合模式生成网络

组合模式生成网络（CPPN）最初被提出，用于将几何抽象直接嵌入发育编码中（Stanley，2006）。这种表示已用于体素优化问题（Auerbach 和 Bongard，2010；Barthet 等，2022）。CPPN 是由多种数学函数组合而成的网络，通过对基底进行逐单元的简单查询来生成表型。基底为表型分配坐标，这些坐标可以构成网格，也可以是机器人系统中各个电机的位置。这种编码已用于创建二维和三维像素／体素网格，例如生成三维物体（Clune 和 Lipson，2011）。解码定义如下：

$$\mathcal{D} : p(x, y) = G(x, y), \text{where } G \text{ a neural graph} \tag{3}$$

神经图 G 为表型中的每个像素或体素返回一个值（图6）。组合模式生成网络通常使用增强拓扑神经进化（NEAT）进行进化，NEAT 是一种进化图对象结构的神经进化方法（Stanley andMiikkulainen, 2002），不过也可以使用任何神经进化算法。

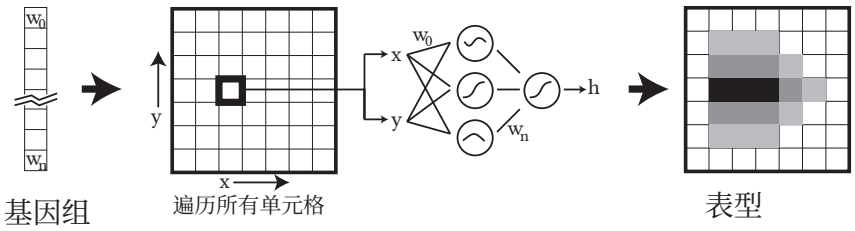

图 6：*CPPN 编码*：固定神经网络的权重和激活函数由基因组编码。

对于网格式表型，网格单元的（x，y）坐标作为查询输入，而建筑模块的高度由CPPN决定。NEAT会扩展神经网络图，因此由于搜索空间的维度可变，不适用于本文的分析。我们不使用NEAT，而是为该图采用固定架构，通过包含一个或两个隐藏层以及预配置数量的隐藏神经元来控制搜索空间的维度。基因组编码了这个固定神经网络图的权重，该图由 l 层和 n 个神经元组成，这些神经元的激活函数从高斯函数、双曲正切函数、Sigmoid函数、正弦函数、余弦函数、恒零、恒一和阶跃函数中随机指定。在较小的概率 p下，变异算子会随机选择一个新的激活函数，或向权重添加一个从标准差为 σ的正态分布中抽取的小值。

2.6 元胞自动机

发育编码，如元胞自动机（CA），由冯·诺伊曼（Neumann，1966）提出，由一个规则集组成，并经过若干次迭代逐步发展。这一时间因素决定了对象的状态能够基于邻域规则发展多长时间。在元胞自动机中，齐次晶格网格中的细胞可以取有限数量的给定状态，这些状态会根据与细胞邻域相关的简单规则发生变化。常用的是摩尔邻域（Moore，1962），它由一个细胞在前一时间步的状态及其全部八个邻居组成。扩展版本允许选取距离细胞超过一个切比雪夫距离的邻居（Ranjan Nayak等，2013）。解码过程包括对表型中的所有细胞进行 t 迭代，其中 t 表示发展过程的长度（图 7）。在每次迭代中，对于每个细胞，下一次迭代中该细胞的值根据其自身及其邻居的旧值计算如下：

$$\mathcal{D} : p(x, y)^{t+1} = \Sigma_{j-1}^{j+1} \Sigma_{i-1}^{i+1} (w_{i,j} \cdot p(i, j)^t) \tag{4}$$

$w_{i,j}$ 在基本元胞自动机中，权重通常是用于检查邻居处于开启还是关闭状态的二值规则，并据此确定单元格的新值。最著名的变体是康威的生命游戏（Gardner，1970），其中包含有关邻居单元格存活或死亡数量的规则。连续值权重已被用于神经元胞自动机方法（Nichele 等，2017；Mordvintsev 等，2020）。
新兴模式难以预测，因为它们会随时间演变，并由局部交互决定。研究主要有两条路线：以目标导向的方式使用元胞自动机，或以开放式探索的方式使用元胞自动机（Herr 和 Ford，2016）。转移规则集的效果难以预测，但这或许有利于质量多样性算法的探索特性。

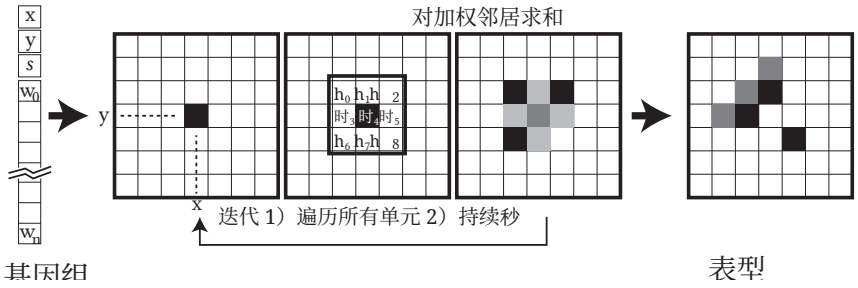

图 7：元胞自动机编码 ：摩尔邻域中相邻元胞影响的权重被编码在基因组中。该 表型经过固定次数的迭代后形成。

本研究采用一种简单的神经细胞自动机设置。首先，将基因组编码的位置（x，y）处的种子设为 1。为了充分表达表型，按照以下迭代发育过程确定细胞的高度值。大小为 $n \times n$，$n \geq 3$ 的扩展摩尔邻域包含一些权重，这些权重用于在方程 4 中计算细胞在下一时间步的值。对所有细胞应用该方程，迭代 t 个时间步。变异算子以较小的概率 p 将种子位置的 x 和 y 坐标移动 1。摩尔邻域掩模的权重按照正态分布进行改变。

2.7 其他编码

其他编码也已应用于类似的问题空间，但不在本文的讨论范围内。最常用的其他编码是基于程序文法的编码，它可以使用计算机辅助设计（CAD）程序中的命令链或命令树来创建对象（Wonka 等，2003）。基于图的编码也可用于基于预定义的 CAD 构建模块集合来规划设计（Keshavarzi 和 Rahmani-Asl，2021）。鉴于这些由软件驱动的编码具有专有性质及其许可模式，本文不予考虑，但未来可在更全面的分析中加以探讨。

深度学习为生成式设计带来了重大突破。变分自编码器（VAE）（Kingma and Welling，2014）等数据驱动生成模型可用于从数据中学习编码（Gaier et al.，2020）。尽管这些数据驱动生成模型能够将后期设计阶段的知识转移到早期阶段，但它们依赖大量训练数据，而这些数据通常并不公开。因此，规模较小的建筑事务所、社区和项目尤其无法依赖此类技术，并可能因此更加依赖私有化模型。不过，质量多样性可用作预计算方法，生成多样化的训练数据集。随后，生成模型会产生高质量的解（Gaier et al.，2020），同时也能遵循设计空间约束（Bentley et al.，2022）。预计算可以解决进化设计中通常会遇到的计算时间过长问题。

系统（Janssen 等，2022）。
鉴于上述原因，非数据驱动且可公开获取的表示技术仍备受关注。因此，比较这些技术中哪些能够生成最多样且高质量的解集颇有意义。

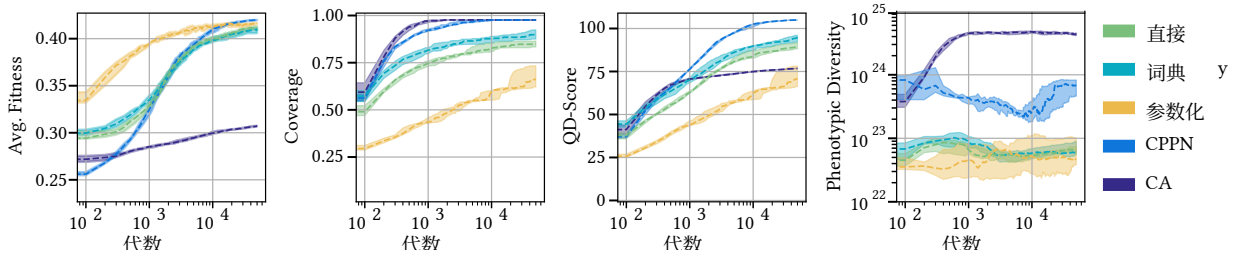

图 8：所有表示的演化。图中包括我们在 MAP-Elites 中结合多种编码的方法。实验重复 10 次。显示的是中位数（虚线）和第 25/75 百分位数。

2.8 多编码质量多样性

我们还评估在质量多样性中采用混合表示是否能提升档案的多样性。多编码质量多样性包含以不同方式编码、并在表型层面相互竞争的物种。尽管经典物种划分的基础是个体只能通过交叉与同种成员繁殖，但在这项初步分析中，我们完全不使用交叉机制。由于 MAP-Elites 不使用交叉操作，因此在 MAP-Elites 中使用多种编码十分直接。每种编码都有自己的变异算子，解仅根据其表型特征和适应度进行比较。初始种群中来自所有编码的个体数量相同。

3 评估

为回答第1节提出的研究问题，我们对比第2.2–2.6节所述的直接和间接表示。为此，每种表示都运行 MAP-Elites 十次，每次从包含 100 个个体的初始种群出发，运行 50,000 代，每代生成十个子代。我们使用第2节所述的代理适应度 函数来确定解的质量。我们通过覆盖率 指标衡量档案的填充程度，即已填充网格单元所占的百分比。*QD* 分数（Pugh等人，2015）对档案中所有单元的适应度值求和，已成为衡量使用固定网格的 QD算法性能的标准指标。不过，该分数取决于问题领域所使用的特征空间，无法体现解的全部多样性。为衡量表型多样性，我们计算档案中表型之间所有成对距离的总和。使用 $L0.1$-范数衡量个体之间的距离，可使多样性指标在高维空间中更加稳健（Wang等人，2016）。

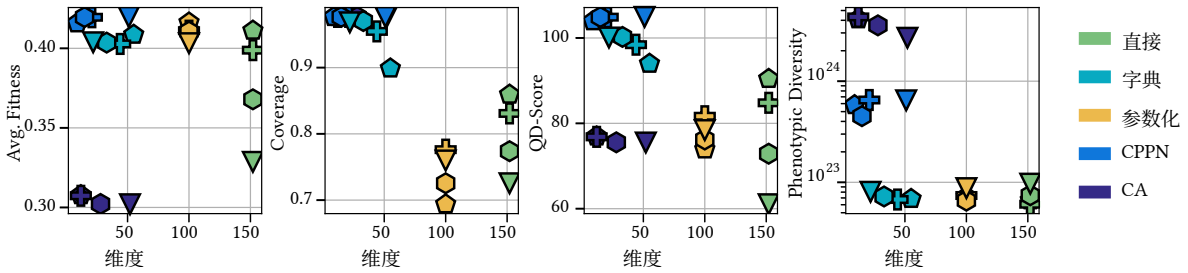

图9：四个指标与编码维度之间的关系。不同的形状表示每种编码的不同参数化方式。请参阅附录中的E节，以获取参数值。

此外，表征的维度通过计算该表征所提供的自由度数量来衡量。我们还评估表征是否能够找到相似的解，或是在解空间中找到不同的区域。最后，如第2.8节所述，我们将表征进行组合，以确定这对QD性能的影响。采用双样本t检验来确定解集的适应度、覆盖率、QD 分数和表型多样性之间的显著性。

3.1 超参数

编码的各种超参数通过使用网格搜索进行参数扫描来确定。对于每种编码，每种表征均运行MAP-Elites十次，每次运行25,000代，初始种群为100，并在每代生成十个子代。根据帕累托支配关系，将平均适应度和表型多样性排列到各个前沿中。随后，从第一个帕累托前沿开始，选出四种最佳超参数配置。超参数值可从代码仓库中获取。

3.2 质量与多样性表征

每种表征的演化过程如图 8所示。初始化时，各表征的质量差异很大。在最初的 20,000 代中，参数化编码的适应度优于所有其他表征。此后，在运行过程中，CPPN 超过参数化编码，找到质量最高的解。直接编码和基于字典的编码的性能相似，这是意料之中的，因为它们能够表达相同的形状。它们的适应度水平未达到参数化

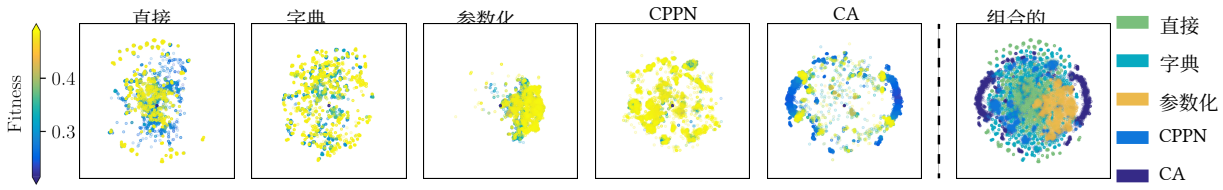

图 10：使用 t-SNE 对所有编码的表型空间进行可视化。编码按适应度进行颜色编码，并在右侧合并显示它们能够到达表型空间的哪些部分。

以及 CPPN。元胞自动机的适应度在所有编码中最低。档案覆盖率一开始的排序则基本相反：元胞自动机立即优于其他编码，不过 CPPN 最终达到相同水平。直接编码仅在覆盖率方面优于参数化编码，而表现不及其他间接编码。

QD 分数展现了更完整的情况，因为覆盖率和适应度都被纳入了考量。CPPN 编码在档案的最大部分区域中找到了性能最高的解，尽管在超越其他编码之前，耗时接近 1,000 代。元胞自动机编码在质量方面表现不佳，但较大的档案覆盖率在一定程度上弥补了这一点。直接编码和基于字典的编码落后于CPPN，但表现优于元胞自动机编码和参数化编码。图右侧展示了表型多样性。元胞自动机编码和CPPN 编码在表型上显然更加多样。参数化编码、直接编码和基于字典的编码之间的多样性相近。编码示例见图 1。

进行了双样本t检验，以确定结果的显著性。请参阅补充材料，查找所有成对的显著性检验。编码之间的大多数比较在统计上均达到 $p < 0.05$。

3.3 表征的维度

当绘制适应度、覆盖率和 QD 分数与编码的维度之间的关系时，图9中呈现出一种规律。表现最佳的编码往往具有更低的维度，无论是在适应度方面还是覆盖率方面。CPPN 和元胞自动机编码的表型多样性要高得多，二者的表现都优于其他编码，而其他编码的表现则相近。有些编码比其他编码对超参数化更加敏感。直接编码的表现就是一个例子，它对超参数最为敏感。根据所使用的超参数，基于字典的编码能够达到与 CPPN 和元胞自动机相近的覆盖率。然而，该编码似乎不像直接编码那样敏感，因为这四个点之间的距离没有那么大。间接编码似乎对超参数化基本不敏感，因此更易于使用。

解集的适应度似乎并不太依赖于维度。然而，对于大多数低维编码，档案的覆盖率、QD 分数和表型多样性都更高。

3.4 表型空间中的不同区域

不同的编码可能能够到达表型空间的不同部分。图10展示了所有编码产生的所有解在二维相似性空间中的分布，该空间通过 t-分布随机邻域嵌入（t-SNE）（Maaten 和Hinton，2008）计算得到。该嵌入基于所有编码的全部10次重复实验中的所有解计算得出。尽管结果不一，但一些间接编码确实能够到达表型空间的不同部分。虽然元胞自动机编码在空间的左侧和右侧找到了非常不同的解，但这些解大多也是性能较低的解。值得注意的是，大多数编码确实可能找到明显不同且通常性能较高的解。

3.5 多编码质量多样性

由于这些编码找到了截然不同的解，一个悬而未决的问题是，在质量多样性中使用多种编码是否有意义。表 1将单一编码质量多样性运行的性能指标与多编码质量多样性进行了比较。使用多种编码时，质量多样性能够达到与使用组合模式生成网络的最佳单一编码运行相近的 QD分数。不过，表型多样性并没有那么高，当然也不及元胞自动机运行，尽管后者找到的主要是性能较低的解。

问题在于，多重编码质量多样性是否仅依赖CPPN来达到这一结果。为回答这一问题，我们分析种群中编码所占的比例。图11展示了一个示例档案，其中

	平均适应度	覆盖率	QD 分数	表型多样性
直接编码	0.411 ± 0.004	0.859 ± 0.042	90.430 ± 3.749	$6.653 \times 10^{22} \pm 1.044 \times 10^{22}$
基于字典的编码	0.409 ± 0.005	0.899 ± 0.038	93.980 ± 2.971	$6.864 \times 10^{22} \pm 2.365 \times 10^{22}$
参数化编码	0.416 ± 0.001	0.694 ± 0.092	73.911 ± 9.650	$7.182 \times 10^{22} \pm 6.032 \times 10^{22}$
组合模式生成网络	$\mathbf{0.420} \pm 0.0$	$\mathbf{0.977} \pm 0.0$	$\mathbf{104.937} \pm 0.075$	$6.465 \times 10^{23} \pm 2.603 \times 10^{23}$
元胞自动机	0.307 ± 0.001	$\mathbf{0.977} \pm 0.0$	76.832 ± 0.244	$\mathbf{4.322 \times 10^{24} \pm 4.243 \times 10^{23}}$
多重编码	0.414 ± 0.002	$\mathbf{0.977} \pm 0.0$	103.504 ± 0.496	$1.281 \times 10^{23} \pm 0.625 \times 10^{23}$

表 1：性能 的单编码与多编码质量多样性。报告的是 10 次实验的均值和 σ s.

CPPN 已经消失。基于字典、参数化和元胞自动机的编码各自占据了档案中的一个连续区域。

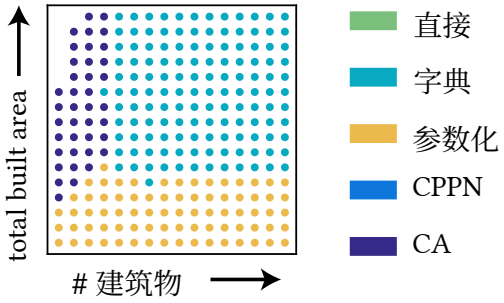

图 11：多编码质量多样性档案中的编码分布示例 .

我们研究了所有 10 次运行中种群内各类编码所占的比例（图 12）。MAP-Elites 初始化时，每种编码的个体数量相同，每种编码 20 个。然而，个体可能被分配到同一个生态位，从而导致较弱的个体被移除。经过种内或种间竞争后，初始化完成时各类编码所占的比例已不再相同。从图中可以看出，在最初约 100 代中，档案的大部分解来自直接编码。之后，基于字典的编码、参数化编码和元胞自动机编码逐渐占据主导。有趣的是，当我们将所有编码合并到单个质量多样性档案中时，在所有编码中表现最佳的 CPPN，在 100 代后大多被移出档案。这可能是由于它在进化早期表现不佳。由于我们不会将编码重新注入档案，一旦某种编码消失，就无法再回归。

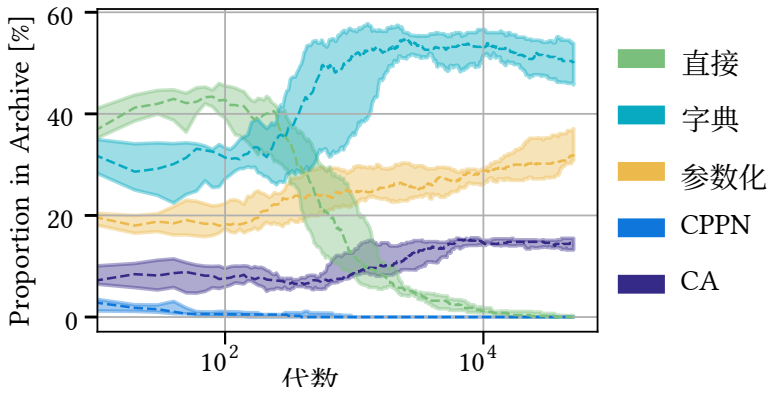

图 12：多编码质量多样性中各编码所占比例。图中显示中位数（虚线）以及第 25/75 百分位数。

不过，有两次运行中 CPPN 在 50,000 代之后仍然存活（见表2）。其中一次运行中，它占据了档案的一半以上。这为我们提供了一些证据，表明我们要么需要保护某些在搜索初期可能更难以发展的物种，要么需要从经典进化算法中开发其他合适的方法来应对这一问题。

#	CA	CPPN	直接	参数化	字典
1	19,2%	0%	0%	31.2%	49.6%
2	13.2%	4.4%	2.8%	22.8%	56.8%
3	12.8%	0%	0%	32.8%	54.4%
4	16.4%	0%	0%	40.4%	43.2%
5	15.6%	0%	0%	30.8%	53.6%
6	14.8%	0%	0%	38.0%	47.2%
7	0%	52.4%	0%	19.2%	28.4%
8	13.6%	0%	0.4%	40.8%	45.2%
9	13.6%	0%	1.6%	30.8%	54.0%
10	15.2%	0%	0%	34.4%	50.4%

Table 2：多编码档案中各编码所占比例。每条线代表一次独立运行（# ）.

3.6 讨论

在第 3.2 节中，我们观察到，在我们的实验中，就质量而言，间接编码有时能够胜过直接编码，但并非总是如此；不过，间接编码在填充档案方面表现更好，具有更高的覆盖率。参数化编码在质量多样性搜索的早期阶段，只能为档案的一小部分找到高质量解，但其 QD 分数低于所有其他编码。在我们定义的特定领域中，我们观察到 CPPN 优于所有其他编码，其次是基于字典的编码和直接编码。在第 3.3 节中，我们观察到，较低的遗传维度往往会产生更高的覆盖率。第 3.4 节中的可视化结果表明，不同的编码可以覆盖表型空间中彼此不同的区域。在第 3.5 节中，我们观察到，多编码质量多样性优于所有其他单一编码；但其表现与 CPPN 相近，尽管它几乎没有使用该编码。

4 结论

与（Tarapore等，2016）所展示的证据相反，我们观察到，在质量多样性中，间接编码可以胜过直接编码。这自然很大程度上取决于该编码能否到达表型空间中的适当区域。较低的编码维度通常能显著提高档案覆盖率，但对解的适应度影响较小。我们观察到，尽管 CPPN 编码在质量和多样性方面都优于其他编码，但在 MAP-Elites 中使用多种编码也能取得相近的结果，即使 CPPN 在与其他编码的竞争中几乎无法存续。多编码质量多样性是进化算法的一种新方法，因为它允许多种定性不同的编码物种 依据其表型（或行为）特征展开竞争。

局限性 尽管使用这些简单的代理模型而非计算流体力学降低了所生成解在实际应用中的真实性，但它们仍可用于比较不同编码，同时减少这项分析所需的计算量。

另一个局限性是，变异算子本身可能使某种编码相对于其他编码占据优势；这种优势可能体现在它们通过在搜索空间中进行较大幅度移动来逃离局部最优的能力上，也可能恰恰体现在进行更细微移动的能力上。

多重编码方法在初始化阶段存在一个弱点，这是由 MAP-Elites 中结构化的档案造成的。在优化过程中后期可能进化出来的解，可能会在初始化阶段或优化早期被较早具备进化能力的编码挤出。物种划分在进化算法中已存在很长时间，但质量多样性或许能带来新的视角。

未来工作 本研究比较了若干常见的表示，但未涵盖一些常见的 CAD 驱动编码；未来工作应将这些编码纳入比较。更严格、全面的分析应涵盖多个领域类别，从机器人控制到形状优化，从而进一步探究在质量多样性中，间接编码还是直接编码更为合适。还应深入分析变异、交叉和选择算子。本研究的观察结果表明，有必要从冗余性、可扩展性和局部性等方面对表征进行更具理论性的分析（Rothlauf，2006），尤其要探究这些因素如何解释并改善多编码质量多样性的表现。例如，由于某些编码（如 CPPN）在初期进化较慢，可以引入保护机制或不同的变异算子。最后，在初始化期间可以使用无结构的档案，以确保解不会立即被覆盖。

本研究表明 在将质量多样性优化应用于问题领域时，应谨慎选择所用的编码。

一种多重编码方法或许能更充分地发挥多种编码各自的优势，同时减少其缺点。由于不同编码在性质上存在差异，我们可以将它们视为真正不同的物种。

致谢

作者感谢 R. 感谢乌多夫·贝伦多夫和贾维德·拉扎克持续不断的开发与支持 BRSU 的计算 g 集群。

资助

计算机硬件由联邦教育与研究部以及北莱茵-威斯特法伦州创新、科学、研究与技术部提供支持（研究资助编号 13FH156IN6）。

参考文献

Auerbach, J. E. 和 Bongard, J. C.（2010）。演化 CPPN 以生长三维实体结构。载于第十二届遗传与进化计算年度会议论文集，第 627–634 页。

Barthet, M.、Liapis, A. 和 Yannakakis, G. N.（2022）。面向《我的世界》建筑生成的开放式演化。*IEEE* 游戏汇刊。

Bentley, P. J.、Lim, S. L.、Gaier, A. 和 Tran, L.（2022）。透过镜子进化：利用变分自编码器学习改进搜索空间。载于自然启发并行问题求解国际会议，第 371–384 页。施普林格。Chen, S.、Montgomery, J. 和 Bolufé-Röhler, A.（2015）。衡量维度灾难及其对粒子群优化和差分进化的影响。应用智能，42：514–526。

Clune,J. 和 Lipson,H.（2011）。采用受发育生物学启发的生成式编码演化三维物体。*ACMSIGEVOlution*，5(4):2–12。

Collins,J.、Cottier, B. 和 Howard,D.（2019）。比较用于特定环境机器人组件设计的直接表示与间接表示。载于*2019*年*IEEE*进化计算大会（*CEC*），第2705–2712页。IEEE。

Cully, A.、Clune,J.、Tarapore, D.和 Mouret, J.-B.（2015）。能够像动物一样适应的机器人。《自然》，521(7553)。Dawkins, R.（1982）。《延伸表型》。牛津大学出版社，牛津。Gaier,A.、Asteroth, A. 和 Mouret, J.-B.（2020）。发现用于黑箱优化的表示。载于《*2020*年遗传与进化计算大会论文集》，第103–111页。Gardner, M.（1970）。约翰·康威的新生命游戏的奇妙组合。《科学美国人》，223：20–123。Hagg,A.（2021a）。发现偏好超体积：面向现实世界计算协同创意的交互式模型。博士学位论文，莱顿大学。Hagg, A.（2021b）。使用质量多样性算法进行表型生态位划分。载于《寻找多个解的元启发式方法》，第287–315页。施普林格。Hagg, A.、Wilde, D.、Asteroth,A. 和Bäck, T.（2020）。使用代理辅助表型生态位划分设计气流。载于《自然启发并行问题求解国际会议》，第140–153页。施普林格。Herr, C. M.和 Ford, R. C.（2016）。建筑设计中的元胞自动机：从通用系统到具体设计工具。《建筑自动化》，72：39–45。Janssen,P.、Bui, T. D. P.和 Wang, L.（2022）。Möbius evolver：城市体量设计策略的竞争性探索。载于《城市规划与设计中的人工智能》，第293–321页。爱思唯尔。Kaushik, V.和 Janssen, P.（2013）。进化设计过程——计算胚胎发生中的自适应迭代探索。Keshavarzi, M. 和 Rahmani-Asl, M.（2021）。Genfloor：通过编码树图实现的交互式生成式空间布局系统。《建筑研究前沿》，10(4)：771–786。Kingma,D.P. 和 Welling, M.（2014）。自动编码变分贝叶斯。《统计学》，1050：1。

Maaten, L. v. d. 和Hinton, G.（2008）。使用 t-sne 可视化数据。机器学习研究期刊，9（11月）。Maher, M. L.（2000）。协同进化设计模型。计算机工程，16(3):195–208。Modrak,V.、Pandian, R. S. 和 Semanco, P.（2021）。使用实验设计校准布局设计优化问题中的遗传算法参数。应用科学，11(15):6940。Moore, E. F.（1962）。自我复制的机器模型。载于 应用数学专题讨论会论文集，第 14 卷，第 17–33 页。美国数学学会，纽约。Mordvintsev, A.、Randazzo, E.、Niklasson,E. 和 Levin,M.（2020）。生长型神经细胞自动机。*Distill*，5(2):e23。Neumann, J. v.（1966）。自我复制自动机理论。计算数学，21:745。Nichele, S.、Ose, M. B.、Risi,S. 和Tufte,G.（2017）。CA-NEAT：用于元胞自动机形态发生与复制的进化组合模式生成网络。*IEEE*认知与发展系统汇刊，10(3):687–700。Pugh, J. K.、Soros, L. B.、Szerlip, P.A. 和Stanley, K. O.（2015）。应对质量多样性的挑战。载于*2015* 年遗传与进化计算大会论文集，第 967–974 页。Ranjan Nayak,D.、Kumar Sahu, S.和Mohammed, J.（2013）。使用二十五邻域模型的基于元胞自动机的最优边缘检测技术。国际计算机应用期刊，84(10):27–33。Rechenberg, I.（1973）。进化策略：依据生物进化原理优化技术系统，Frommann–Holzboog。德国斯图加特。Rothlauf,F.（2006）。遗传与进化算法的表征。施普林格。Schwefel, H.-P. 和 Schwefel,H.-P.（1977）。数值优化的进化策略。施普林格。Sobieczky, H.（1999）。参数化翼型与机翼。空气动力学设计方法的最新发展：逆向设计与优化，第 71–87 页。Stanley, K. O.（2006）。利用规律性而不进行发展。载于 *AAAI* 秋季专题讨论会：发展系统，第 49页。Stanley, K.O.和Miikkulainen, R.（2002）。通过扩充拓扑演化神经网络。进化计算，10(2):99–127。Tarapore,D.、Clune, J.、Cully, A. 和 Mouret, J.-B.（2016）。不同编码如何影响MAP-Elites算法的性能？载于 *2016* 年遗传与进化计算大会论文集，第 173–180 页。Wang, H.、Jin,Y. 和 Yao, X.（2016）。多目标优化中的多样性评估。*IEEE* 控制论汇刊，47(6):1510–1522。Wang,L.、Chen, K. W.、Janssen, P. 和 Ji, G.（2020）。实现基于优化的建筑体量设计探索——Rhino-Grasshopper 中无需编码的进化建筑体量设计工具包。Wang,L.、Janssen,P.、Chen,K.W.、Tong,Z. 和Ji,G.（2019）。面向性能的建筑设计探索中的减法式建筑体量设计：日照优化案例研究。可持续性，11(24):6965。Wonka, P.、Wimmer, M.、Sillion, F. 和 Ribarsky,W.（2003）。即时建筑。*ACM* 图形学汇刊（*TOG*），22(3):669–677。Xu,X.、Yin, C.、Wang,W.、Xu, N.、Hong, T.和 Li, Q.（2019）。通过遗传算法驱动的城市街区设计揭示中国干热地区的城市形态与户外舒适度。可持续性，11(13):3683。

附录

显著性检验 适应度

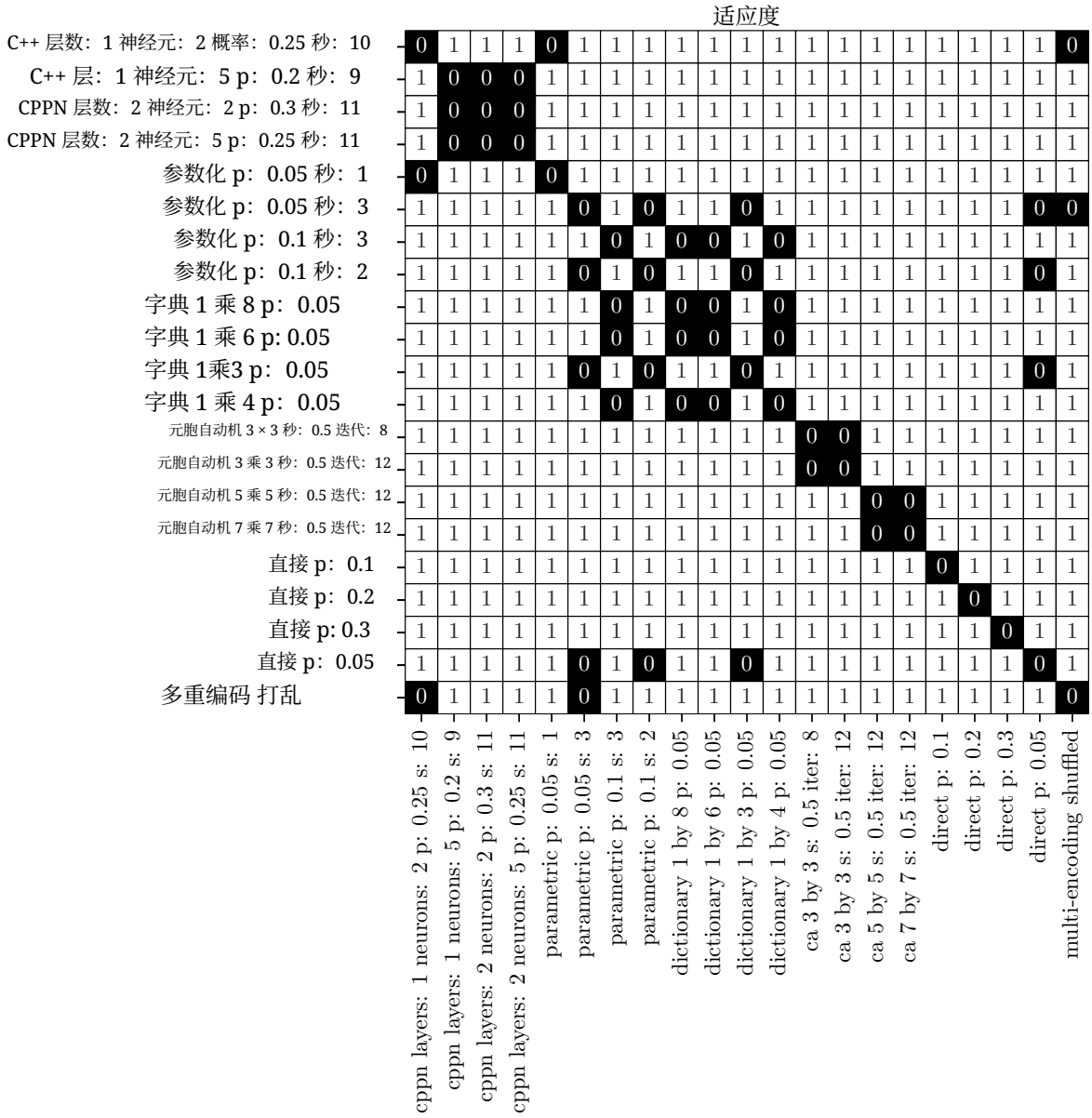

图13：比较编码/超参数的成对适应度值的双样本t检验结果，针对$p < 0.05$。

Significance Tests Coverage

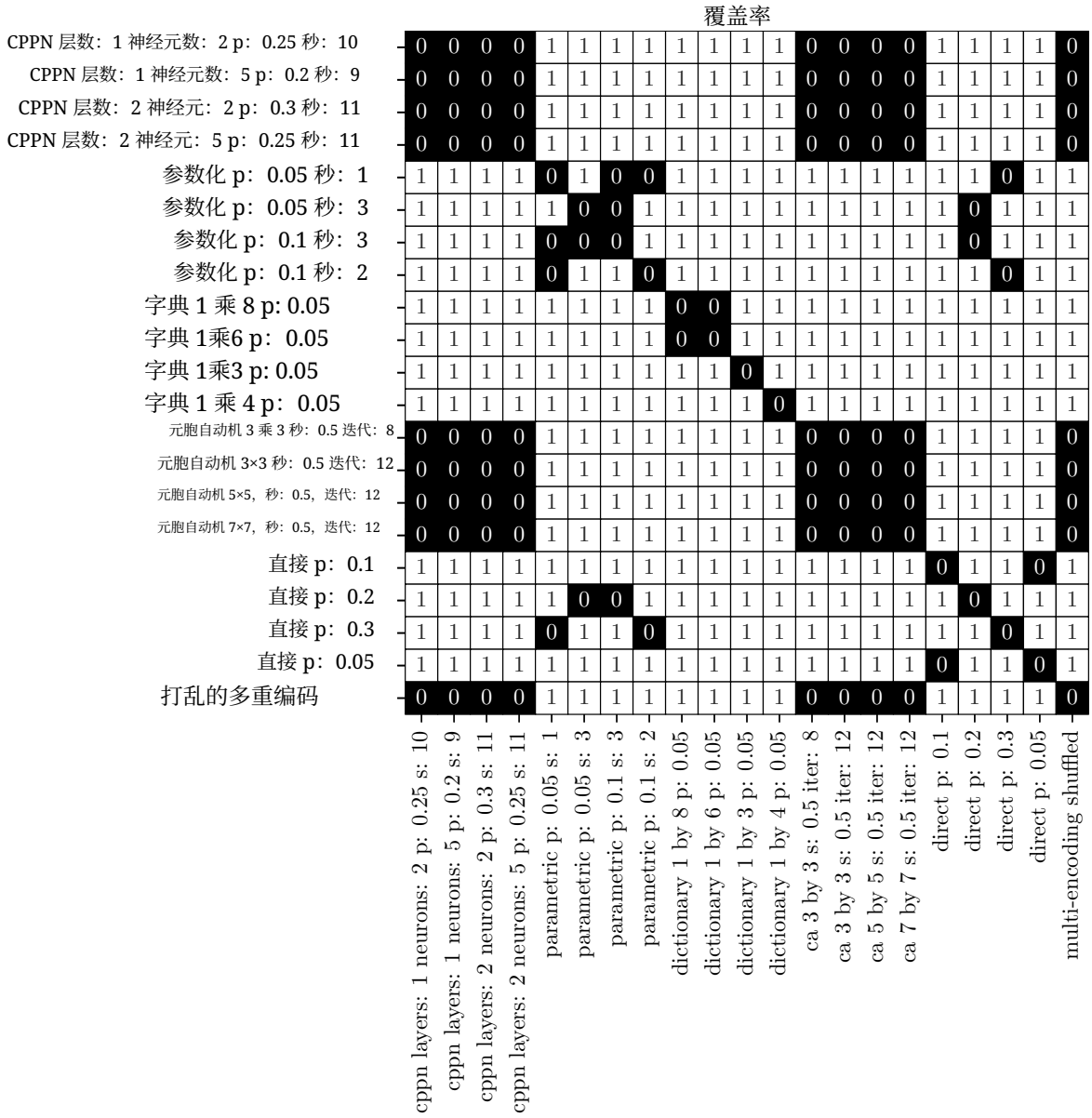

图14：比较 $p < 0.05$的编码/超参数成对覆盖率的双样本t检验结果。

QD 分数显著性检验

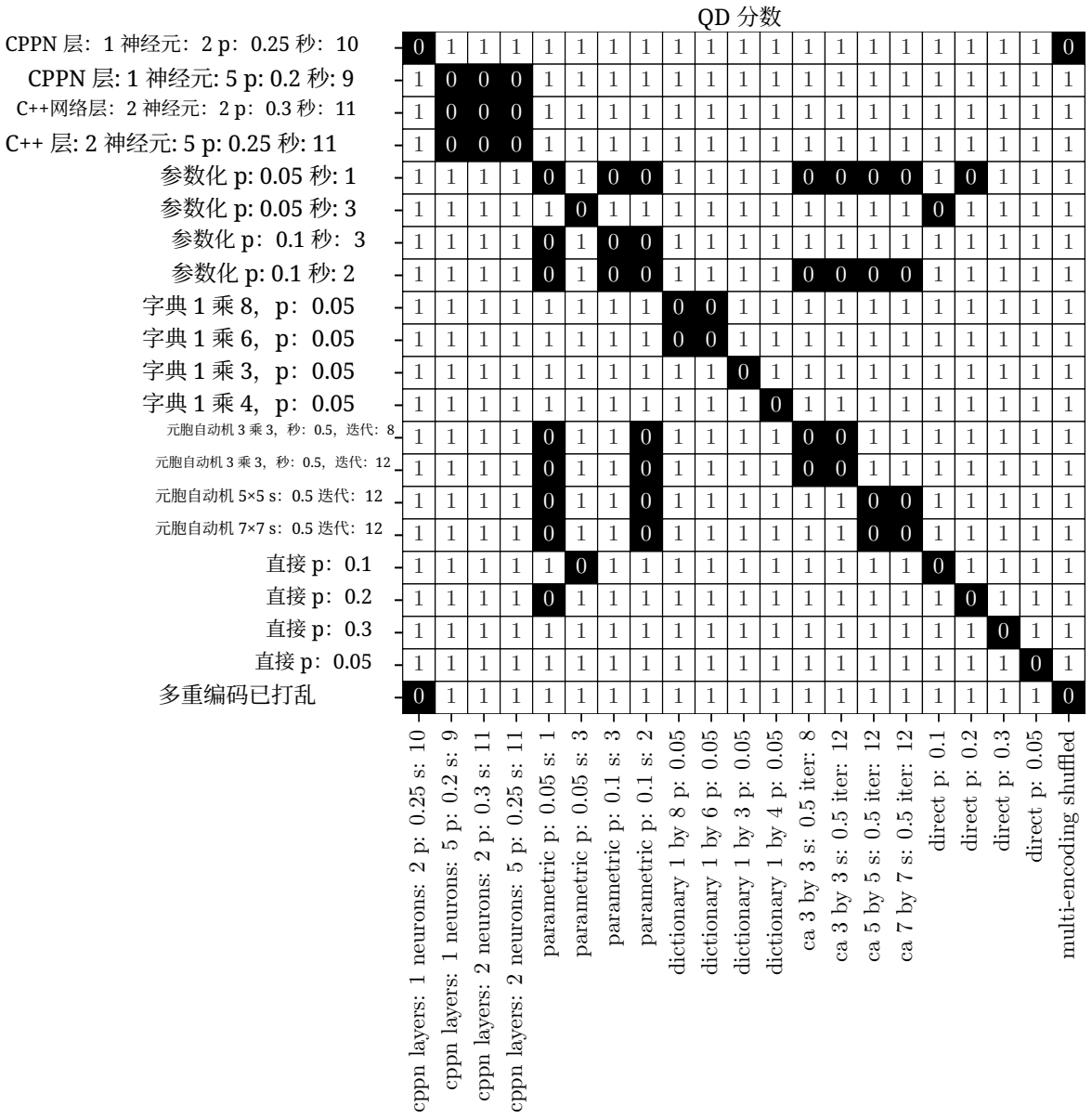

图 15：比较$p < 0.05$的编码/超参数两两 QD 分数的双样本t检验结果。

表型多样性显著性检验

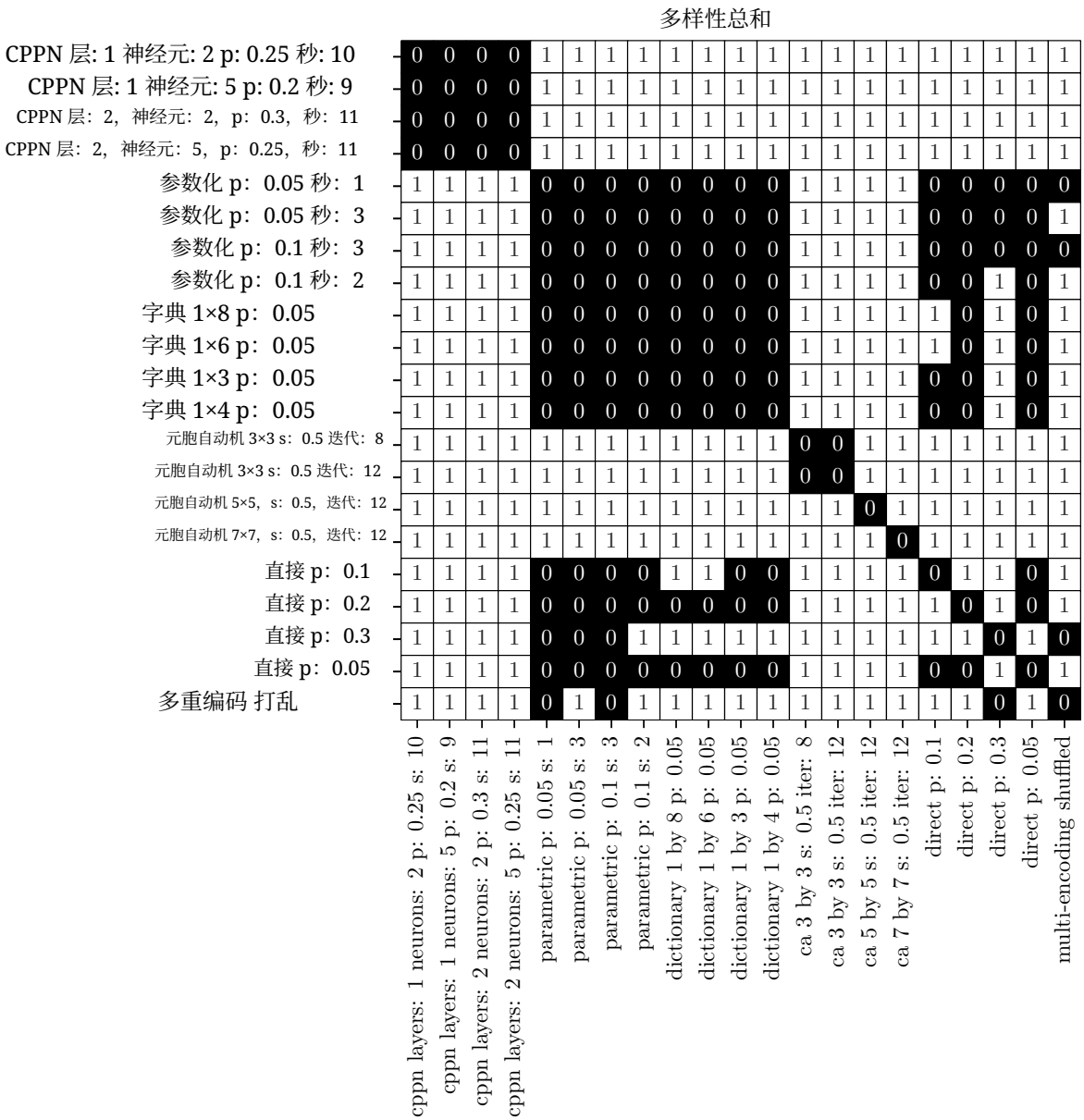

图16：针对 $p < 0.05$，比较编码方式/超参数的成对表型多样性值的双样本t检验结果。

符号说明

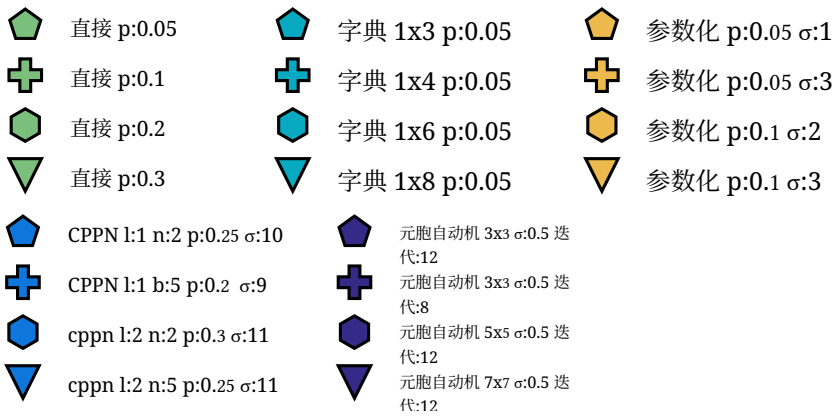

图 17：各编码对应不同参数化方式的说明。

更多编码示例

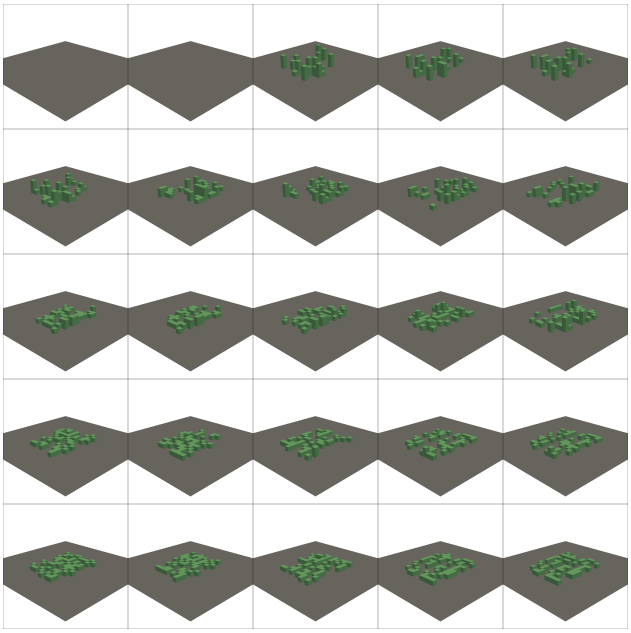

图 1　　8：使用直接编码的随机选定档案，其中包含 25 个代表性解。

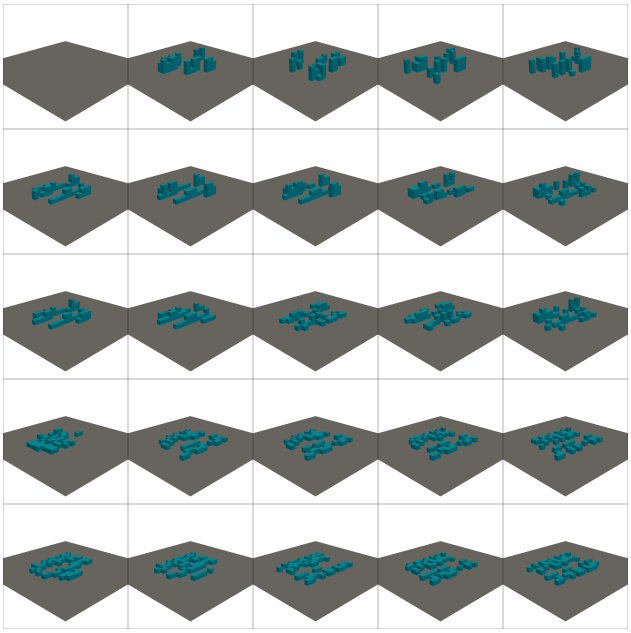

图 19：使用字典编码的随机选取档案，其中包含 25 个代表性解。

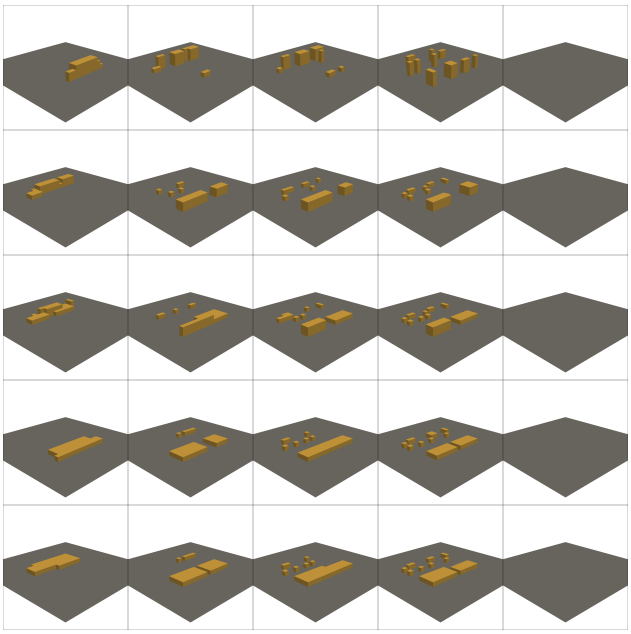

图20：使用参数化编码的随机选取档案，其中包含25个代表性解。

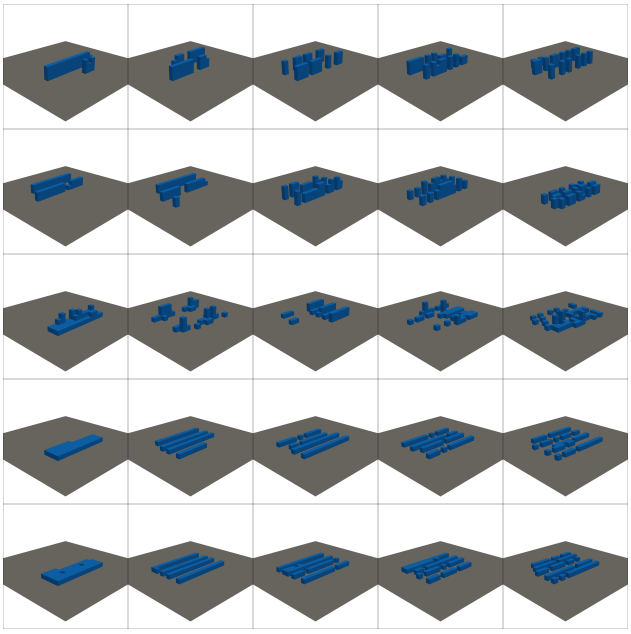

图21：使用CPPN编码的包含25个代表性解的随机选取档案。

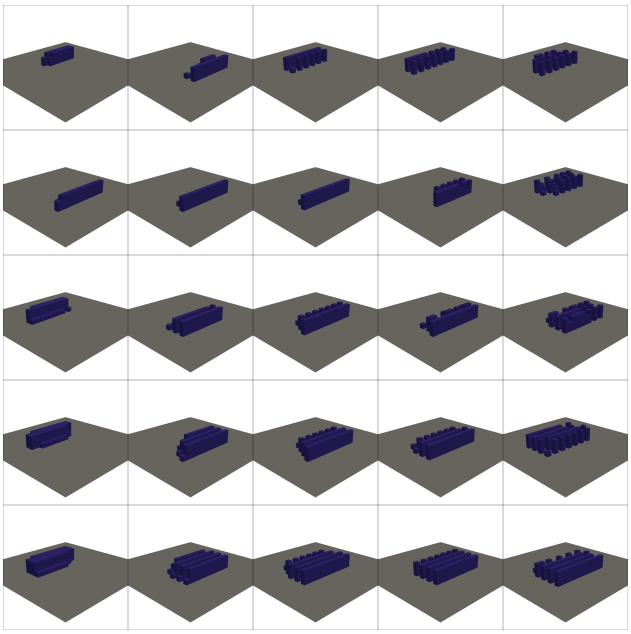

Figure 22：使用细胞自动机编码随机选择的包含25个代表性解的档案 .