arXiv:2304.03672v1 [cs.NE] 7 Apr 2023

别只靠运气：增强不确定领域中质量-多样性解的行为可复现性

Luca Grillotti*
伦敦帝国理工学院
伦敦，英国
luca.grillotti16@imperial.ac.uk

Manon Flageat*
伦敦帝国理工学院
英国伦敦
manon.flageat18@imperial.ac.uk

Bryan Lim
伦敦帝国理工学院
英国伦敦
bryan.lim16@imperial.ac.uk

Antoine Cully
伦敦帝国理工学院
英国伦敦
a.cully@imperial.ac.uk

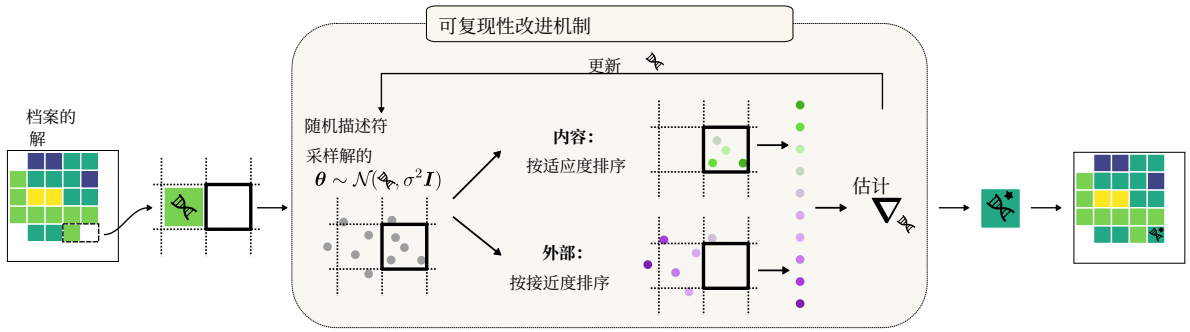

图 1： 档案可复现性改进算法 **(ARIA) 在“完成阶段”中的一次迭代。选取两个相邻单元格，其中一个是空的目标单元，另一个已由一个解填充。随后，** 可复现性改进机制 **以该解为起点，在目标单元中寻找高性能解。经过数步可复现性改进后，我们将得到的优化解添加到目标单元中。**

摘要

质量多样性（QD）算法旨在生成高性能解集合，同时最大限度地提升这些解在给定描述符空间中的多样性。然而，当存在不可预测的噪声时，同一解的适应度和描述符可能会因评估次数不同而出现显著差异，从而导致这些值的估计存在不确定性。鉴于 QD 算法具有精英主义特性，在此类噪声环境中，它们最终往往会产生许多退化解。在本研究中，我们提出了档案可复现性改进算法（ARIA），这是一种即插即用的方法，可提升档案中解的可复现性。我们将其设计为一个独立的优化模块，基于自然进化策略，可叠加运行于任何 QD 算法之上。该模块通过变异解来（1）优化其属于自身利基的概率，并（2）最大化其适应度。我们在多种……场景中评估了该方法的性能。

*两位作者对本研究贡献相同。

允许为个人或课堂使用而免费制作本作品的部分或全部内容的数字或纸质副本，前提是不得为盈利或商业利益而制作或分发副本，且副本首页须包含本声明及完整引用信息。本作品中第三方组件的版权必须得到尊重。其他用途请联系所有者/作者。*GECCO '23*，*2023*年7月*15*日至*19*日，葡萄牙里斯本© 2023 版权所有归所有者/作者所有。ACM ISBN 979-8-4007-0119-1/23/07。
https://doi.org/10.1145/3583131.3590498

任务，包括一个经典优化问题和两个在模拟机器人环境中的高维控制任务。我们表明，我们的算法能使任何给定档案的质量和描述符空间覆盖率至少提升 50%。

CCS 概念

• **计算方法** → **进化机器人学**。

关键词

质量-多样性、不确定领域、神经进化

ACM 参考格式：
Luca Grillotti、ManonFlageat、BryanLim 和 Antoine Cully。2023年。不要只押注运气：提升不确定领域中质量-多样性解的行为可复现性。载于 遗传与进化计算大会（*GECCO '23*），*2023*年7月*15*日至*19*日，葡萄牙里斯本。ACM，纽约，美国纽约州，12页。https://doi.org/10.1145/3583131.3590498

1 引言

质量多样性（QD）算法 [7, 28, 29] 是进化算法，旨在生成多样且性能优异的解集合或档案。这些档案随后可用于各种应用，包括机器人学中的损伤恢复 [1, 2, 5, 22]、电子游戏设计 [8, 15], 以及城市布局配置 [18]。

质量多样性方法依赖于一个评估过程，用于估算给定解的性能；对每个解进行评估时，还会得到一个描述符，用以表征该解相较于其他解的新颖性。大多数质量多样性算法

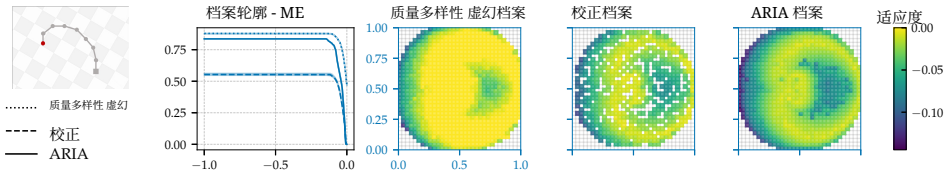

图 2：机械臂任务中由 ARIA-ME 初始化返回的档案轮廓、虚幻档案及相应的校正档案。ARIA 档案展示了经过 ARIA 优化后得到的结果。

的设计都假设这一评估过程是确定性的。这里的确定性是指，对一个解进行评估总会得到相同的结果（适应度和描述符）。然而在现实中，许多情境和应用中的评估过程都存在不确定性。例如，在机器人学中，传感器的测量结果可能带有噪声，向执行器发送指令也可能无法产生精确的预期位移；在这些情况下，向机器人发送指令不一定总能得到相同的结果。领域中存在不确定性的质量多样性设置属于不确定性 QD 框架 [12]。如果质量多样性算法的设计没有考虑这种不确定性，就可能生成“虚幻”的档案，其中许多解的性能和描述符估算有误，并且可能表现出较大的方差 [12]。在这种情况下，我们称这些解的可复现性较低。要使解具有较高的可复现性，其方差必须较低，且描述符和性能的估算必须准确。

在这项工作中，我们提出了档案可复现性改进算法 （ARIA），这是一种能够降低解的方差、即提高其可复现性的方法。它可以作为即插即用工具，用于任何解集，包括由质量多样性和进化策略算法返回的解。该方法依托可复现性改进机制（图 1），利用自然进化策略降低解的描述符方差，同时最大化其适应度。我们利用这一机制，不仅提高档案中现有解的可复现性，还能发现新解，从而提升给定档案的多样性（图 2）。我们在三个经典的不确定性 QD 任务上，将 ARIA 与不同的关注可复现性的基线方法进行比较，结果表明，ARIA 找到的档案在性能和可复现性方面均表现最佳。具体而言，与标准基线方法相比，ARIA 的最佳变体使 QD 分数提高了 30%，并将可复现性至少提升了 20%。

2 背景

2.1 进化策略

在这项工作中，我们打算使用适用于可能具有高维度的问题（包括深度神经进化任务）的进化策略优化算法。Salimans 等人 [30]提出了一种受自然进化策略启发的方法 [32]：他们不直接针对目标函数 $\tilde{o}(\cdot)$进行优化，而是最大化 $\tilde{o}$在高斯分布下的期望：$\mathrm{E}_{\theta\sim\mathcal{N}(\phi,\sigma^2 I)}[\tilde{o}(\theta)]$，其中均值 ϕ 是待优化变量，标准差 σ保持不变。在进化策略中，通常使用基于排名的效用值（u_s），而非目标函数值 [19, 32]。在这种情况下，用于更新参数的梯度可以表示为：$\frac{1}{\sigma}\sum^{N_S}_{s=1} u_s \epsilon_s$，其中 ϵ_s从正态分布中采样。

2.2 质量-多样性算法

质量多样性（QD）算法 [7, 29] 旨在为优化问题寻找一组多样且性能优异的解，这组解也称为“档案”。与常见的进化算法不同，QD 算法认为，解不仅要分配适应度，还要分配用于衡量其新颖性的描述符。MAP-Elites 等常见的 QD 算法 [5, 25] 会将描述符空间离散化为网格。对于网格中的每个单元格，这些算法都旨在找到一个描述符落在该单元格内、且适应度能够使该单元格性能达到最高的解。每次迭代时，它们按如下方式运行：从档案中选取解，并通过随机变化（例如突变、交叉等）对其进行修改；随后评估这些解的适应度和描述符，再将其放回档案。更具体地说，要将一个解加入档案，我们首先检查其描述符所属的单元格。如果该单元格为空，就将此解加入其中；否则，如果该单元格已被另一个解占据，则只保留性能最好的解。通过这些机制，MAP-Elites 及其变体会逐步填充更多单元格，同时提升档案中所存解的性能。

2.3 描述符可复现性最大化的不确定性质量多样性设置

对于某些环境，对同一解进行连续评估可能会得出不同结果。例如，在机器人学中，这通常是由不同的初始条件、传感器噪声或关节松动造成的。在 Flageat 和 Cully [12], 形式化定义的不确定性 QD 设置中，对同一解进行 $\boldsymbol{\theta}$ 连续评估可能会得到不同的适应度和描述符。在此类设置中，由解$\boldsymbol{\theta}$ 得到的适应度 f和描述符 $\boldsymbol{d}$ 可表示为两个分布 $\mathrm{D}(\boldsymbol{\theta})_f$ 和 $\mathrm{D}(\boldsymbol{\theta})_{\boldsymbol{d}}$。考虑将描述符空间划分为若干单元格（$\mathrm{C}_i$），该问题可表述为：对于每个单元格 C_i，我们要寻找一个解 $\boldsymbol{\theta}_i$，使期望适应度 $\mathrm{E}_{f_i\sim\mathcal{D}_f^{(\boldsymbol{\theta}_i)}}[f_i]$ </style>最大化，同时期望描述符 $\mathrm{ED}_d^{(\boldsymbol{\theta}_i}[\boldsymbol{d}_i])$属于单元格 C_i。此外，我们还旨在最小化描述符各维度 j上描述符分量的方差 。换言之，我们考虑以下多目标约束优化问题：

$$\begin{aligned}&\text{For each cell } C_i\text{, find a solution } \boldsymbol{\theta}_i \text{ that:}\\ &\quad\text{maximize } \mathrm{E}_{f_i\sim\mathcal{D}_f^{(\boldsymbol{\theta}_i)}}[f_i]\\ &\quad\text{minimize } \mathrm{Var}_{\boldsymbol{d}_i\sim\mathcal{D}_d^{(\boldsymbol{\theta}_i)}}\left[\boldsymbol{d}_i^{(j)}\right]\ \forall j\\ &\quad\text{subject to } \mathrm{E}_{\boldsymbol{d}_i\sim\mathcal{D}_d^{(\boldsymbol{\theta}_i)}}[\boldsymbol{d}_i]\in C_i\end{aligned}\tag{P1}$$

算法 1 ARIA

输入： 由进化策略或质量多样性算法 Φ 给出的基因型集合；梯度步数 N梯度；每个梯度步的样本数 Ns；样本标准差 σ

2: $\mathcal{P}$expl $\leftarrow \emptyset$,$\mathcal{P}$targ $\leftarrow$划分描述符空间的所有单元格

▷ 可复现性改进阶段

3: **对于** 基因型 $\phi \in \Phi$ **执行**

4: $(\boldsymbol{d}_s)^{N\text{s}}_{s=1} \leftarrow$ 评估基因型 Ns 次并收集描述符

5: $C_j \leftarrow$查找包含平均描述符的单元格 $\boldsymbol{d}$

6: $\boldsymbol{\phi}_j \leftarrow$ 可复现性改进（$\boldsymbol{\phi}$init $= \boldsymbol{\phi}$、Ctarg $= C_j$、 Ngrad、Ns、σ）

7: 将C_j 添加到$\mathcal{P}$expl，并从$\mathcal{P}$targ中移除它

▷ 存档完成阶段

8: **当**$\mathcal{P}$targ 非空**执行**

9: 选择 2 个相邻单元格，使得 Cs $\in \mathcal{P}$expl和 Ctarg $\in \mathcal{P}$targ

10: 与 $\leftarrow$基因型相关的$\boldsymbol{\phi}$的Cෙන්ටර්ඩ්

11: $\boldsymbol{\phi}$targ $\leftarrow$ 可复现性改进($\boldsymbol{\phi}$init $= \boldsymbol{\phi}_{\text{s}}$,$C$targ, Ngrad, Ns, σ)

12: 将 $\boldsymbol{\phi}$targ 添加到目标单元 Ctarg

13: 将 Ctarg 添加到$\mathcal{P}$expl 中，并将其从 $\mathcal{P}$targ 中移除

14: **返回** 所有单元格中的所有基因型

2.4 相关工作

不确定领域中的质量多样性。 随着质量多样性领域不断发展，并逐步应用于神经进化和复杂机器人学应用 [13]，近期文献中开始逐渐关注将质量多样性应用于不确定领域的问题。面对不确定性，质量多样性算法难以估计解的期望质量和新颖性。为解决这一局限，人们提出了多种 MAP-Elites 变体。这些变体要么通过多次采样解来应对——可以动态进行 [12, 21]，也可以不动态进行 [6] ——要么依赖基于先前个体 [11, 12]的更复杂机制。此外，梯度增强型质量多样性方法（如 PGA-MAP-Elites [26]和 MAP-Elites-ES [4]）在应对不确定性方面也展现出良好前景。其基于梯度的机制能够利用信息生成比传统质量多样性方法 [10]更具可复现性的解。然而，所有这些先前工作都依赖于修改 MAP-Elites 主算法，以直接生成更好的性能估计并鼓励产生可复现的解。相比之下，我们的方法 ARIA 提出了一种即插即用机制，能够改进任意给定 QD 算法的结果，包括那些未考虑任务不确定性的算法。

将质量多样性与进化策略结合。 以往已有多项研究提出在 MAP-Elites 中融入进化策略。值得注意的是，CMA-ME 算法 [16] 提出使用 CMA-ES 为 MAP-Elites 生成后代。与 ARIA 类似，CMA-ME 优化的是不同于原始任务适应度的目标，例如档案改进分数。与我们的工作更为接近的是，MAP-Elites-ES [4]使用 Salimans 等人 [30]提出的自然进化策略 [32]在 MAP-Elites 中生成后代，并针对任务适应度或解的新颖性进行优化。这两种方法通过在质量多样性算法的变异过程中融入进化策略机制，改进了 MAP-Elites；而 ARIA 则提出将进化策略机制作为第二步，应用于所选的质量多样性算法之上。此外，ARIA 使用的目标函数不同于 CMA-ME 和 MAP-Elites-ES 所采用的目标函数。事实上，ARIA 的目标函数旨在优先优化解的可复现性。再者，CMA-ME 和 MAP-Elites-ES 都可以用作 ARIA 的初始化算法。

以往的质量多样性定向改进工作。 不同研究分支提出了多种机制，用于从质量多样性档案中选择高效的优化起点。MAP-Elites-ES [4] 采用一种有偏的亲代选择机制；该机制会选择那些倾向于产生适应度更高或新颖性更强的后代解的亲代。类似地，Go-Explore [9] 通过选择相关单元格进行探索来解决高难度探索任务。在 ARIA 的最后阶段，其选择机制会迭代地选取位于已发现描述符空间边界处的一个解，并对其进行优化，以便在相邻的空单元格中找到新解，如图 1 所示。

3 方法

在本节中，我们首先介绍不确定性质量多样性问题P1的一种替代性问题表述。然后，我们介绍档案可复现性改进算法（ARIA），这是一种即插即用的方法，可以优化一组解，从而生成一个包含多样、高性能且具有可复现性的解的档案。最后，我们将详细介绍ARIA所依赖的可复现性改进机制。

3.1 问题定义

如第 2.3 节所述，不确定性QD问题可以形式化为一个多目标问题：最小化描述符方差，同时最大化每个单元格内的期望适应度（见问题P1）。然而，直接求解这一优化问题在计算上颇具挑战。我们不再最小化描述符空间中的方差，而是提出最大化每个解的描述符属于其指定单元格的概率。由此，问题P1变为：

$$\begin{aligned} &\text{For each cell } C_i, \text{ find a solution } \boldsymbol{\theta}_i \text{ that:} \\ &\qquad \text{maximize } \mathrm{E}_{f_i \sim \mathcal{D}_f^{(\boldsymbol{\theta}_i)}}\left[f_i\right] \\ &\qquad \text{maximize } \mathrm{P}\left(\boldsymbol{d}_i \in C_{\text{targ}}\right) \\ &\qquad \text{subject to } \mathrm{E}_{\boldsymbol{d}_i \sim \mathcal{D}_d^{(\boldsymbol{\theta}_i)}}\left[\boldsymbol{d}_i\right] \in C_i \end{aligned} \tag{P2}$$

问题P1中所表达的描述符方差与新的概率目标相关。通过最大化 $\mathrm{P}\left(\boldsymbol{d}_i \in \mathrm{C}_i\right)$，我们可以最小化协方差矩阵迹的一个下界（证明见附录 A）。

3.2 ARIA 概述

在本研究中，我们提出了档案可复现性改进算法 (ARIA)，以解决问题P2。ARIA 的输入是一组解；这组解可以是 QD 算法返回的档案，也可以是进化策略算法返回的单个优化解。因此，ARIA 是一种即插即用的方法，可应用于 QD 或进化策略算法返回的任意解集；它会返回一个包含高性能、多样化且高度可复现解的档案。ARIA 的概要分为两个连续阶段，具体内容均在算法 1 中详述：

(1) The可复现性改进阶段：通过使用 可复现性改进机制，提高作为输入提供的解的可复现性。

(2) 档案补全阶段：使用相同的可复现性改进机制，在仍为空的档案单元格中寻找高性能且可复现的解。

3.2.1 可复现性改进阶段。可复现性改进阶段旨在提高作为算法输入的解的可复现性。为此，对于输入集中的每个解，我们对其重新评估 N 次，并计算其平均描述符；平均描述符定义为对同一解进行 N 次重新评估时所观测到的描述符的平均值。随后，我们检查哪个单元格 Ctarg 包含该平均描述符。然后，我们使用可复现性改进机制，针对单元格 Ctarg 对该解的性能和可复现性进行优化。该可复现性改进机制包括使用一种进化策略对该解进行 N 个梯度步骤的优化；更多细节如下所述。这些步骤在算法 1 的第 3–7 行中有详细说明。

*3.2.2*存档完成阶段。 在可复现性改进阶段结束时，网格中仍有一些尚未被覆盖的单元格。例如，如果没有输入解的平均描述符最终落入某个单元格，那么在前一阶段结束时，该单元格仍然是空的。因此，在改进所有初始解的可复现性之后，我们有两组互补的单元格：Pexpl和 Ptarg。Pexpl包含截至目前已被可复现性改进过程用作目标的所有单元格。本阶段的目的是寻找填充剩余单元格的控制器，以探索 Ptarg。为此，我们按照图 1 所示，逐步探索 Ptarg 中的剩余单元格。

在每次迭代循环中，我们首先选择一对满足三个条件的单元格 Cs 和 Ctarg：(i) Cs $\in$ Pexpl，(ii) Ctarg $\in$ Ptarg，且 (iii) Cs 和 Ctarg 相邻。然后，我们从 Cs 中取出初始解 $\boldsymbol{\phi}$init，并使用可复现性改进来优化其适应度以及落入 Ctarg 的概率。最后，我们收集这一优化过程得到的解 $\boldsymbol{\phi}$togo。请注意，没有任何因素能保证优化后解的平均描述符$\boldsymbol{\phi}$togo落在目标单元格 Ctarg[1]中。当描述符单元格不可达，或者可复现性改进机制过早结束时，就可能出现这种情况。无论如何，我们都会将 $\boldsymbol{\phi}$ togo 添加到目标单元格 Ctarg；并将 Ctarg 添加到已探索单元格集合 Pexpl，同时将其从 Ptarg 中移除。我们重复上述步骤，直至没有更多单元格可供探索。所有这些细节均见算法 1 第 8 行及之后的内容。

3.3 可复现性改进机制

如上一节所述，可复现性改进机制以待优化的解和目标单元 Ctarg 作为输入。该机制旨在优化解的期望适应度以及其属于 Ctarg 的概率，同时满足以下约束：其平均描述符属于 Ctarg（见问题P2）。为了解决针对特定目标单元 C = Ctarg 及其关联质心 $\boldsymbol{c}$ 的约束优化问题，我们采用了 Salimans 等人 [30]，在第 2.1 节和算法 2 中详细介绍的进化策略。

[1]我们的所有指标和图表都考虑到了这一点。换句话说，如果平均描述符 $\boldsymbol{\phi}$togo 不在 Ctarg 中，那么我们的任何指标和图表都不会将其纳入 Ctarg。

算法 2 可复现性改进

输入： 初始基因型 $\boldsymbol{\phi}$init，单元格 Ctarg，梯度步数 Ngrad，数量每个梯度步的样本数 Ns，标准差采样 σ
$\boldsymbol{\phi} \leftarrow \boldsymbol{\phi}$init
对于步骤 = $1 \rightarrow N$梯度 **执行**
  对于 $s = 1 \rightarrow N_s$ **执行**
    样本 $\boldsymbol{\epsilon}_s \sim \mathrm{N}(\mathbf{0}, \boldsymbol{I})$
    $f_s, \boldsymbol{d}_s \leftarrow$ 基因型的随机评估 $\boldsymbol{\theta}_s = \boldsymbol{\phi} + \sigma\boldsymbol{\epsilon}_s$
  $(u_s) \leftarrow$ 基于排名的效用值取决于它们在目标函数中的排序 $o_C$targ（公式 1）
  梯度_估计值 $\leftarrow \frac{1}{\sigma} \sum_{s=1}^{N} \mathrm{s} u_s \boldsymbol{\epsilon}_s$
  $\boldsymbol{\phi} \leftarrow$对 $\boldsymbol{\phi}$ 使用梯度_估计值执行梯度上升步骤
返回最终基因型 $\boldsymbol{\phi}$

该进化策略算法通过以下步骤迭代更新解 $\boldsymbol{\phi}$：(1) 遵循高斯分布，在搜索空间中对邻近解进行采样和评估；(2) 按偏好顺序对这些解进行排名；(3) 根据估计的基于排名的梯度更新 $\boldsymbol{\phi}$（见第 2.1 节）。为了提高 $\boldsymbol{\phi}$的可复现性，我们按以下方式对解进行排名：

- 所有评估后的描述符位于该单元格内的解，其排名都高于位于单元格外的解。
- 在描述符位于单元格外的解中，根据其与单元格中心的距离进行排名：距离越近，排名越高。
- 在描述符位于单元格内的解中，根据其适应度进行排名：适应度越高，排名越高。

更正式地说，这等价于最大化以下目标函数，其满足上述所有排序规则：[2]

$$\widetilde{o}_C(\boldsymbol{\theta}) = o_C(f, \boldsymbol{d}) \quad \text{with } f \text{ and } \boldsymbol{d} \text{ sampled from } \mathcal{D}_f^{(\boldsymbol{\theta})}, \mathcal{D}_d^{(\boldsymbol{\theta})}$$
$$= \begin{cases} f & \text{if } \boldsymbol{d} \in C \\ f_{\min} - \|\boldsymbol{d} - \boldsymbol{c}\|_2 & \text{if } \boldsymbol{d} \notin C \end{cases} \tag{1}$$

这种目标函数结构已在进化计算文献中被提出，用于处理约束优化问题 [3, 19]。请注意，函数 $\widetilde{o}(\cdot)$ 是随机的，而函数 $o(\cdot)$ 是采样适应度和描述符的确定性函数。此外，若假设适应度有界，则可以证明，最大化 $\widetilde{o}_C$targ $(\cdot)$ 的期望值，就能最大化问题P2的两个目标的下界，即期望适应度以及属于某个描述符单元格的概率（证明见附录 A）。

4 实验设置

4.1 指标

4.1.1 期望适应度（*EF*）和（校正后的）*QD* 分数。期望适应度（EF） [13] 刻画了解决方案相对于任务的平均性能。如问题 P1 和 P2 所示，它是我们的算法明确要最大化的量之一。为了估算某个解决方案的期望适应度，我们会对其重新评估 M

[2]对于此公式，我们假设 f有界，因为这样可以简化记法。不过，这一假设并不苛刻，因为如果适应度无界，则可将 $f$min 设为迄今为止遇到的最小适应度。

次，并对得到的适应度取平均值。

$$\mathrm{EF}(\boldsymbol{\theta}) = \mathrm{E}_{f\sim\mathcal{D}_f^{(\boldsymbol{\theta})}}[f] \approx \frac{1}{M}\sum_{m=1}^{M} f_m \quad \text{with } f_m \sim \mathcal{D}_f^{(\boldsymbol{\theta})} \tag{2}$$

（修正后的）QD分数 [13, 29] 刻画了档案的质量和多样性。它对应于档案中所包含解的所有期望适应度之和。为处理负值，在求和之前，首先将所有期望适应度归一化到 0 到 1 之间。

4.1.2 负描述符方差（NDV）和方差分数（V- Score）。 NDV [12] 刻画了描述符样本的离散程度。从数学上看，它对应于描述符样本协方差矩阵的迹的相反数。如果一个解决方案具有较高的NDV 分数，即NDV 分数接近0，则意味着其样本描述符（平均而言）集中在其均值附近。该指标来源于我们在原始不确定性QD问题中最小化的方差目标函数（见问题P1）。我们使用以下无偏估计量对其进行评估（其中$\boldsymbol{d} = \frac{1}{M}\sum_{m=1}^{M}\boldsymbol{d}_m$）：

$$\begin{aligned}\mathrm{NDV}(\boldsymbol{\theta}) &= -\sum_j \mathrm{Var}_{\boldsymbol{d}\sim\mathcal{D}_d^{(\boldsymbol{\theta})}}\left[\boldsymbol{d}^{(j)}\right] \\ &\approx \frac{-1}{M-1}\sum_{m=1}^{M}\left\|\boldsymbol{d}_m - \overline{\boldsymbol{d}}\right\|_2^2 \quad \text{with } \boldsymbol{d}_m \sim \mathcal{D}_d^{(\boldsymbol{\theta})}\end{aligned}$$

方差分数（V-Score）是NDV对应的质量多样性-Score。它不考虑归一化的期望适应度，而是将档案中每个单元格的归一化 NDV 分数相加。

4.1.3 属于平均描述符单元格的概率与概率分数（P分数）。 单元格归属概率定义为：解的描述符属于其指定单元格的概率。该指标刻画了解的可复现性；它对应于我们的算法所最大化的概率目标函数（见问题P2）。给定一个解 $\boldsymbol{\theta}$，我们首先确定平均描述符 $\boldsymbol{d}$ 所属的单元格C_i；随后通过重新评估其描述符 M 次，并计算最终落入该单元格的样本所占比例，来估计 $\mathrm{P}(\boldsymbol{d}\in\mathrm{C}_i)$：

$$\mathrm{P}(\boldsymbol{d}\in C_i) \approx \frac{1}{M}\sum_{m=1}^{M}\varphi(\boldsymbol{d}_m \in C_i) \quad \text{with } \boldsymbol{d}_m \sim \mathcal{D}_d \text{ and } \overline{\boldsymbol{d}} \in C_i$$

where $\varphi(\cdot)$ works as an Iverson bracket: $\varphi(P) = \begin{cases}1 \text{ if } P \text{ is true}\\ 0 \text{ if } P \text{ is false}\end{cases}$

与 QD 分数和 V-Score 类似，我们考虑概率分数（P 分数）。该 P 分数等于最终档案中每个解所获得的概率值之和。

4.1.4 档案轮廓。 档案轮廓是一种全面可视化所得档案适应度分布的方法 [13, 14]。更准确地说，对于给定的适应度阈值f_{thres}，档案轮廓对应于具有更高期望适应度的解的档案单元C 的数量。在我们的实验中，我们通过将上述数量除以单元格总数对其进行归一化。这样，我们估计出由解填充的网格单元比例

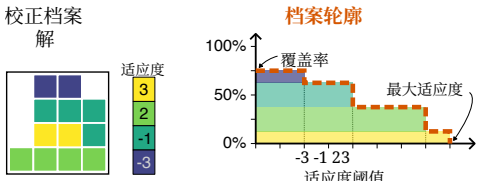

图 3：展示如何从校正档案（左侧）推导出档案轮廓（右侧的橙色虚线）。我们还说明了如何从档案轮廓图中获取档案覆盖率和最大适应度。

基线	Reprod*	适应度	描述符
MOME-R	✓	$\begin{cases}\overline{f} = \frac{1}{N_r}\sum_{s=1}^{N_r} f_s \\ - \quad - \quad \text{NDV}\end{cases}$	$\overline{\boldsymbol{d}} = \frac{1}{N_r}\sum_{s=1}^{N_r}\boldsymbol{d}_s$
ME-Sa-R	✓	$\overline{f} - NDV^{**}$	$\overline{\boldsymbol{d}}$
ME-Sa	✗	$\overline{f}$	$\overline{\boldsymbol{d}}$
MAP-Elites	✗	f	$\boldsymbol{d}$

* 基线也会针对可复现性进行优化吗?

** 为使两个项的数值处于同一数量级，先对其进行归一化，再求和。**表 1：所研究基线方法的特征。**

适应度高于 f_{thres}：

$$\textsc{ArchiveProfile}(f_{\text{thres}}) = \frac{1}{N_C}\sum_{\text{cell } C}\varphi\left(f_C \geq f_{\text{thres}}\right)$$

其中 $\varphi(\cdot)$是艾弗森括号，如第4.1.3节所述。

档案轮廓呈现出档案的若干重要属性。例如，其在 y 轴上的最左侧取值对应档案的总覆盖率；而曲线与 x 轴的交点则表示档案中找到的最高适应度（示意见图 3）。

4.2 校正档案

QD 算法返回的档案可能包含退化解，其描述符可能与期望描述符存在显著差异。因此，在此类档案中，解被放入的单元格可能并不是其期望描述符所对应的单元格；我们将这些档案称为：虚幻档案。为缓解这一问题，我们的结果仅以校正档案 [11, 13, 21]的形式呈现。给定一个 QD 算法返回的档案或基因型集合，所有基因型都会被重新评估 M 次，并根据其平均描述符 $\boldsymbol{d} = \sum_{s=1}^{M}\boldsymbol{d}_s$被放入一个名为“校正档案”的新档案中；在每个单元格中，我们只保留具有最佳估计期望适应度的解（见公式 2）。虚幻档案及其校正档案之间的主要差异如图 2 所示。

请注意，在本研究中，上文详述的所有档案指标均为“校正”的 [13]：它们仅应用于校正档案。因此，在讨论 QD 分数、NVS 分数和 P 分数时，我们省略“校正”前缀。

4.3 ARIA 变体

ARIA 可以接受任意一组基因型作为输入。如上文所述，在 ARIA 中，这组基因型会经过可复现性改进阶段，随后被用作在档案的其余部分中寻找新的可复现解的基础。为了研究初始基因型集合的影响，我们使用不同类型的输入来初始化 ARIA。相应的变体如下：

• **ARIA- ME 初始化**，其输入是 MAP-Elites 返回的档案。这样，我们便能用表现出色、覆盖描述符空间不同区域的控制器来初始化 ARIA 算法。

• **ARIA-PGA初始化**，它以 PGA-ME 返回的档案 [26] 作为输入。尽管这种环境仅与 QD-RL 任务 [31], 兼容，但预计当解参数化高维深度神经网络控制器时，它将返回更优的档案。

• **ARIA- ES 初始化**运行进化策略算法，以最大化单个解的性能。最大化的目标仅为期望适应度；并且我们使用与之前相同的优化过程（见第 2.1 节）。

此外，我们还提出了一种名为"线性可复现性改进机制"（**线性 RIM**）的新变体，它优化的目标函数与 ARIA 不同。通过这一变体，我们旨在说明 ARIA 的可复现性改进机制所采用的基于约束的目标函数同样重要。具体而言，线性 RIM 使用的目标函数等于公式 1 中两个（归一化）组成部分之和。其余优化流程保持不变；与 ARIA- ME 初始化类似，它以 MAP-Elites 生成的档案作为输入。

4.4 基线方法

我们考虑与 ARIA 进行比较的第一个基线方法是**MAP-Elites** 算法，该算法在每次解评估中仅采样一个描述符和一个适应度。这个较弱的基线方法未考虑问题中的不确定性。因此，MAP-Elites 返回的虚幻档案与其对应的校正档案可能存在显著差异（见图 2）。

We intend to al所以比较我们的变量iants of ARIA to several alg为不确定性QD问题设计的算法；这些算法详述如下，其特点总结于表1。

• MAP-Elites Sampling（**ME-Sa**）的工作方式与 MAP-Elites 相同，只是它会为每个解采样多个适应度和描述符，并利用它们计算平均适应度和平均描述符。这样一来，ME-Sa就能更准确地估计解的期望适应度和描述符；同时，ME-Sa 返回的档案预计会与其校正档案更加相似。 [21]

• 考虑可复现性的 MAP-Elites采样（**ME-Sa-R**）与ME-Sa 类似，不同之处在于，会修改每个解的适应度，以考虑可复现性问题。ME-Sa-R的适应度在预期任务适应度之上增加了一项，用于惩罚 NDV 分数较高的解（见表 1）。

• 考虑可复现性的多目标 MAP-Elites（**MOME-R**）。由于待解决的问题最初是一个多目标问题（见问题P1），我们还提议研究多目标 MAP-Elites（MOME）变体所获得的结果 [27]。MOME-R 使用的两个适应度指标，是 ME-Sa-R 基线所用目标函数中的两个项。与 MOME 的开创性工作一样，我们将帕累托前沿长度设为 50。在这种情况下，MOME-R 算法无法与我们的方法直接比较，因为它返回的档案可能包含远多于其他所有算法的解。事实上，MOME-R 的档案中每个单元格最多可以存储 50 个解。为了使这种方法具有可比性，在分析时，我们只选取每个单元格中使 ME-Sa-R 目标函数达到最大值的解（见表 1）。

4.5 任务

我们比较了所有算法在不确定性质量多样性设置中常用的三个任务上的性能 [12]：带噪声的 8 自由度（DoF）机械臂任务，以及全向蚂蚁方向和单向 Walker 任务。

4.5.1 噪声 *8* 自由度机械臂。机械臂任务由一个平面机器人机械臂组成，我们控制它到达不同的最终位置 [7]。解由连续执行器$\boldsymbol{\theta} = (\theta_i)_{i \in [1..8]}$的角位置组成。适应度鼓励具有均匀角度集的解，并向其中加入一些高斯噪声，使其评估次数具有不确定性。此外，描述符对应于受到一些高斯噪声扰动的末端执行器最终位置（x_E, y_E）。适应度和描述符可表示如下，其中 $\sigma_f = \sigma_d = 0.01$：

$$f(\boldsymbol{\theta}) = -\mathrm{Var}\,(\theta_i) + \mathcal{N}\left(0, \sigma_f^2\right) \quad \boldsymbol{d}(\boldsymbol{\theta}) = \begin{pmatrix} x_E \\ y_E \end{pmatrix} + \mathcal{N}\left(\mathbf{0}, \sigma_d^2 \boldsymbol{I}\right) \quad (3)$$

4.5.2 全向蚂蚁。全向蚂蚁任务旨在寻找高维控制器，使四足机器人能够到达多样化的最终位置。该任务符合质量多样性强化学习设置 [13, 31]；它可建模为马尔可夫决策过程（MDP），任务适应度等于整个回合内奖励的非折扣总和。在全向蚂蚁任务中，奖励鼓励解避免过早死亡，同时惩罚过高的能耗 $f = \sum_{t=1}^{T} r_{survive} - r_{energy}$ with $T = 100$。此外，描述符对应于机器人的最终位置：$\boldsymbol{d} = (x_T,\ y_T)$。我们使用 QDax 库 [24]，中实现的全向蚂蚁任务，该库本身依赖于 Brax 中的蚂蚁环境 [17]。学习得到的控制器是闭环深度神经网络策略，包含两个大小为 64 的隐藏层，并在每个时间步输出施加于机器人关节的扭矩。在此任务中，噪声施加于初始关节位置和速度。

4.5.3 单向 *Walker*。单向 Walker 任务旨在寻找多种尽可能快速向前移动的方式。与全向蚂蚁任务类似，该任务符合质量多样性强化学习的设定，我们同样采用基于 Brax 的 QDax 实现。此外，所考虑的噪声和控制器与蚂蚁任务相同。不过，其适应度和描述符与蚂蚁任务不同。具体而言，此处的奖励还包含一项鼓励向前移动的奖励：

$f = \sum_{t=1}^{T} r_{forward} + r_{survive} - r_{energy}$，其中 $T = 1000$。此处的描述符用于表征每条腿触地时间所占的比例：$\boldsymbol{d} = \frac{1}{T}\sum_{t=1}^{T} (C_1(t)\ C_2(t))$，其中如果腿 i 在时间步 t 触地，则 $C_i(t)$ 等于 1，否则等于 0。

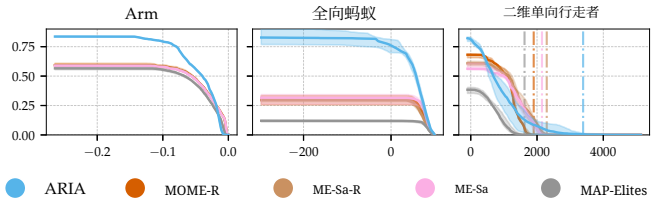

图 4：ARIA 和其他基线方法的档案分布。为清晰起见，每张图中仅展示 ARIA 的一个变体。从左到右，图中展示的 ARIA 变体依次为：ME 初始化、ME 初始化和 PGA 初始化。粗线表示中位数，阴影区域对应四分位距。行走者任务中的竖线表示最大适应度的中位数。

4.6 实现细节

对于每个研究任务，描述符空间均为二维，并离散化为一个 32×32 网格。作为输入提供给 ARIA-ME 初始化和 ARIA-PGA 初始化的档案，分别通过 2×10^6 次和 5×10^5 次评估生成。在可复现性改进函数的每次迭代中，机械臂和全向蚂蚁任务会生成 $N_S = 2{,}048$ 个样本；行走者任务则选用 $N_S = 1{,}440$ 个样本，因为其计算开销通常更大。我们还采用镜像采样，因此可复现性改进机制每次迭代分别执行 4,096 次和 2,880 次评估。此外，机械臂、蚂蚁和行走者任务所抽取样本的标准差分别为 $\sigma = 0.005$、0.02 和 0.01。最后，可复现性改进机制在机械臂和蚂蚁任务中执行 $N_{\text{grad}} = 100$ 个梯度步；在行走者任务中执行 $N_{\text{grad}} = 75$ 个梯度步。

MOME-R、ME-Sa-R 和ME-Sa 变体使用128 的批次大小，每个解重新评估 $N_r = 32$ 次，因此每次迭代的总采样规模为 [12] 4,096。因此，我们也将该数值用作 MAP-Elites基线方法的批次大小。所有这些基线方法的总预算均等于ARIA 变体执行的最大评估次数，即机械臂和蚂蚁任务为 4.3×10^8，行走者任务为 2.3×10^8。

在分析时，每个解决方案都会被重新评估 $M = 1{,}024$ 次，以计算档案轮廓。此值用于所有指标。

我们的所有实现均基于 QDax 框架 [24],，但可复现性改进机制除外，该机制改编自 Brax [17] 中的进化策略实现。每项实验均在 NVIDIA RTX A6000 上运行；使用该硬件，ARIA 分别花费 20 分钟、10 小时和 18.5 小时完成机械臂、Ant 和行走者任务。我们对每个变体进行了 10 次重复实验；并使用 Wilcoxon 秩和检验评估比较结果的统计显著性，同时采用 Holm-Bonferroni 校正 [20]。为便于复现实验结果，我们将代码及其依赖项存储在 Singularity 容器 [23], 中，并与代码一同发布于：https://bit.ly/aria-gecco。

5 结果

在所研究的三个任务中，ARIA 显著改善了初始 QD 算法生成的校正档案的档案轮廓。具体而言，ARIA 在所有任务中都将校正档案的覆盖率至少提高了 50%（$p < 2 \times 10^{-4}$，图 2 和5)。请注意，我们并不打算达到虚幻档案的结果，因为这类档案包含许多经评估的适应度和描述符与其期望适应度和描述符存在显著差异的解；因此，没有任何信息能够说明是否可以获得这类档案。有趣的是，在蚂蚁任务上，ARIA 获得的覆盖率甚至优于虚幻档案的覆盖率（$p < 2 \times 10^{-4}$）。此外值得注意的是，在行走者任务上，与ARIA-ME初始化相比，ARIA-ME 的校正档案中适应度高于 900 的个体更多；这一现象可能是由于ARIA在优化性能之前会优先优化可复现性，这一点也得到了表 2 中较高的方差分数以及附录 C 所提供的扩展分析的佐证。总体而言，这些结果表明，ARIA 不仅提高了任意 QD 档案中所含解的可复现性，还能够改善其内容，并且有时还能提高其覆盖率。我们的结果视频见：https://sites.google.com/view/gecco2023-aria/。

ARIA-ME初始化和ARIA-PGA 初始化实现了比其他未使用可复现性改进机制的基线更好的可复现性，即更高的 V 分数和 P 分数（表 2 和图 6，$p < 0.03$）。这可以通过ARIA明确最大化的目标函数来解释，该目标函数在优化性能之前优先关注可复现性。事实上，第3.3节引入的目标函数优先选择位于单元格中的解，而不是不在单元格中的解。此外，可复现性改进机制中的进化策略可以扩展到高维深度神经网络，从而使 ARIA 更容易明确地优化可复现性。

如果比较各种 ARIA- ME 初始化所获得的结果，我们会发现，每种变体的性能都与其初始化相关。例如，在蚂蚁任务中，PGA 初始化档案的性能不佳，导致结果质量低下（图 5）。相反，依赖于更优校正档案的 ME 初始化变体，能够取得更高的 QD 分数和档案覆盖率（$p < 0.03$）。令人意外的是，ARIA 的 ES 初始化变体取得的 QD 分数优于其 PGA 对应变体（表 2）。然而，在该任务中，PGA 初始化仍然能够提供更多可复现档案，这很可能归因于其内部优化过程，正如文献所示 [10, 26]。

在所有任务中，最佳QD分数均由 ARIA的某个变体获得（见表 2），其档案分布表现出更高的覆盖率（图 4）。此外，在行走者任务中，档案轮廓表现出更高的最大适应度。ARIA 的某些变体所获得的较差 QD 分数（例如，机械臂任务中的 ES初始化和行走者任务中的ME 初始化）可能是由其初始化方式导致的。事实上，从单个高性能个体开始探索（如ARIA- ES 初始化所采用的方式），可能不足以在机械臂任务的搜索空间中找到所有精英个体；而行走者任务中解的高维性阻碍了 ME 初始化找到高性能个体，这似乎对行走者任务不利。

可复现性提升机制所采用的基于约束的目标函数也很重要。事实上，无论是在性能还是可复现性方面，线性 RIM 始终无法超越 ARIA-ME。在机械臂和蚂蚁任务中，与 ARIA-ME 相比，它的质量多样性、方差和概率分数都更低（表 2，$p < 8 \times 10^{-3}$）。在行走者任务中，线性 RIM 达到了一个

算法	机械臂任务			全向蚂蚁任务			单向 Walker 任务		
	QD 分数	V-Score	P 分数	QD 分数	V-Score	P 分数	QD 分数	V-Score	P 分数
ARIA - ME 初始化	**722.41**	426.83	**653.37**	**765.69**	**818.14**	**453.94**	81.04	776.81	561.96
ARIA - PGA 初始化	✗	✗	✗	679.06	756.64	414.12	180.72	**839.87**	**634.06**
ARIA - ES 初始化	587.30	**428.22**	647.37	581.22	613.61	279.80	**248.14**	701.43	466.58
线性 RIM	704.68	394.23	542.58	610.28	619.30	341.06	115.13	642.80	373.22
MOME-R	538.62	303.57	258.03	287.90	268.79	47.77	175.75	692.74	284.29
ME-Sa-R	538.19	304.37	258.94	283.39	279.24	61.34	185.68	613.97	174.67
ME-Sa	534.06	301.37	253.61	307.26	251.20	33.40	168.70	554.26	123.61
MAP-Elites	506.34	288.63	318.59	117.91	93.65	13.44	66.87	357.52	19.75

表 2：各算法在所研究的三个任务中取得的中位数（校正后）QD 分数、方差分数和概率分数。请注意，机械臂任务不是 QD-RL 任务 [31]，，因此 PGA-ME 无法直接应用于该任务。

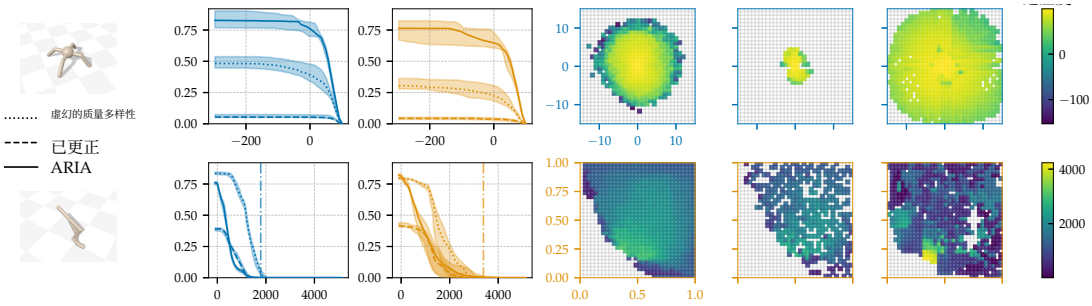

图 5：（第 1 和第 2 列）ARIA 的 ME 初始化和 PGA 初始化在蚂蚁任务与行走者任务上的档案分布；（第 3 至第 5 列，从左到右）QD 算法返回的虚幻档案、与之对应的校正档案，以及从 ARIA 优化后获得的档案（所选档案是在 10 次重复实验中取得中位数 QD 分数的档案）。在行走者任务中，竖线表示 ARIA 返回的档案所达到的最大适应度中位数。

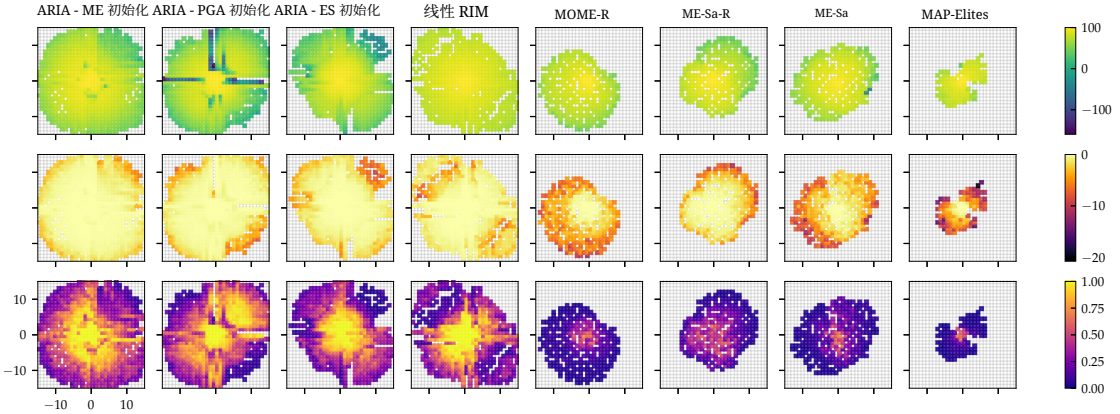

图 6：在蚂蚁任务中，各算法返回的档案中期望适应度、NDV 分数以及属于某个单元格的概率分布。所选档案是在 10 次重复实验中达到中位数 QD 分数的档案。

相比ARIA-ME初始化，其质量多样性分数更高，但其方差和概率分数仍显著更低（$p < 2 \times 10^{-4}$）。

6 讨论

在本研究中，我们提出了 ARIA，这是一种即插即用算法，旨在：（1）提升 QD 档案中现有解的性能和可复现性；（2）利用这些解，在校正档案的空单元格中寻找新解。在所研究的三项任务中，我们证明 ARIA 显著提升了校正档案的整体性能、描述符方差和覆盖率。我们还证明，ARIA 可以同样应用于多种 QD 算法之上。

然而，我们的工作存在一些局限性。首先，它需要大量的评估次数；可以研究其他优化方法，以提高样本效率。此外，“档案补全”阶段中单元格的选择方式仍可改进；目前，单元格的选择并未考虑其解决方案的性能或可复现性。最后，我们仅考虑了相对于描述符的可复现性；研究如何同时最小化适应度方差也具有重要意义。

致谢

本研究得到工程与自然科学研究委员会（EPSRC）资助的 EP/V006673/1 项目 REcoVER 的支持。

参考文献

[1] Maxime Allard、Simón C Smith、Konstantinos Chatzilygeroudis 和 Antoine Cully。2022。用于在线损伤恢复的分层质量-多样性。载于《遗传与进化计算大会论文集》。58–67。[2] Konstantinos Chatzilygeroudis、Vassilis Vassiliades 和 Jean-Baptiste Mouret。2018。面向机器人损伤恢复的无重置试错学习。《机器人与自主系统》100（2018），236–250。[3] Carlos A Coello Coello。2002。进化算法中使用的理论与数值约束处理技术：现状综述。《计算方法与应用力学及工程》191，11-12（2002），1245–1287。[4] Cédric Colas、Vashisht Madhavan、Joost Huizinga 和 Jeff Clune。2020。将 MAP-Elites 扩展至深度神经进化。载于《2020年遗传与进化计算大会论文集》。67–75。[5] Antoine Cully、Jeff Clune、Danesh Tarapore 和 Jean-Baptiste Mouret。2015。能够像动物一样适应的机器人。《自然》521，7553（2015），503–507。[6] Antoine Cully 和 Yiannis Demiris。2018。采用无监督描述符的分层行为库。载于 GECCO 2018——《2018年遗传与进化计算大会论文集》。计算机协会，纽约州纽约市，美国，69 76 1804 07127 – 。arXiv: 。Antoine Cully 和 Yiannis Demiris。[7] 2018。《质量与多样性优化：统一模块化框架》。《IEEE进化计算汇刊》22，2（2018年4月），245–259。arXiv:1708.09251[8] Sam Earle、Justin Snider、Matthew C Fontaine、Stefanos Nikolaidis 和 Julian Togelius。2022。用于关卡生成的多样化神经元胞自动机的展现。载于《遗传与进化计算大会论文集》。68–76。[9] Adrien Ecoffet、Joost Huizinga、Joel Lehman、Kenneth O Stanley 和 Jeff Clune。2021。先返回，再探索。《自然》590，7847（2021），580–586。[10] Manon Flageat、Felix Chalumeau 和 Antoine Cully。2022。不确定领域中用于神经进化的 PGA-MAP-Elites 实证分析。《ACM进化学习汇刊》（2022）。[11] Manon Flageat 和 Antoine Cully。2020。在噪声领域中使用深度网格实现快速稳定的 MAP-Elites。载于《人工生命大会论文集》。麻省理工学院出版社，273–282。[12] Manon Flageat 和 Antoine Cully。2023。不确定质量-多样性：不确定领域中质量-多样性的评估方法与新方法。arXiv 预印本 arXiv:2302.00463（2023）。[13] Manon Flageat、Bryan Lim、Luca Grillotti、Maxime Allard、Simón C Smith 和 Antoine Cully。2022。对用于强化学习的神经进化质量-多样性算法进行基准测试。arXiv 预印本 arXiv:2211.02193（2022）。[14] Matthew Fontaine 和 Stefanos Nikolaidis。2021。可微质量-多样性。《神经信息处理系统进展》34（2021），10040–10052。[15] Matthew C. Fontaine、Ruilin Liu、Ahmed Khalifa、Jignesh Modi、Julian Togelius、Amy K. Hoover 和 Stefanos Nikolaidis。2020。在生成对抗网络的潜在空间中展现马里奥场景。arXiv: 2007 05674 [ ] 。cs.AI [16] Matthew C Fontaine、Julian Togelius、Stefanos Nikolaidis 和 Amy K Hoover。2020。用于快速展现行为空间的协方差矩阵自适应。载于《2020年遗传与进化计算大会论文集》。94–102。[17] C. Daniel Freeman、Erik Frey、Anton Raichuk、Sertan Girgin、Igor Mordatch 和 Olivier Bachem。2021。Brax——用于大规模刚体仿真的可微物理引擎。http://github.com/google/brax[18] Theodoros Galanos、Antonios Liapis、Georgios N Yannakakis 和 Reinhard Koenig。2021。ARCH-Elites：面向城市设计的质量-多样性。arXiv 预印本 arXiv:2104.08774（2021）。[19] Nikolaus Hansen。2016。CMA进化策略：教程。arXiv 预印本 arXiv:1604.00772（2016）。[20] Sture Holm。1979。一种简单的序贯拒绝多重检验程序。《斯堪的纳维亚统计学杂志》（1979），65–70。[21] Niels Justesen、Sebastian Risi 和 Jean-Baptiste Mouret。2019。通过自适应采样处理噪声领域中的 MAP-Elites。载于《遗传与进化计算大会伴随论文集》。121–122。[22] Rituraj Kaushik、Pierre Desreumaux 和 Jean-Baptiste Mouret。2020。用于机器人学中基于行为库的在线适应的自适应先验选择。《机器人与人工智能前沿》（2020），151。[23] Gregory M Kurtzer、Vanessa Sochat 和 Michael W Bauer。2017。Singularity：支持计算资源迁移的科学计算容器。《公共科学图书馆·综合》12，5（2017），e0177459。[24] Bryan Lim、Maxime Allard、Luca Grillotti 和 Antoine Cully。2022。通过大规模并行实现加速质量-多样性。《机器学习研究汇刊》（2022）。https://openreview.net/forum?id=znNITCJyTI[25] Jean-Baptiste Mouret 和 Jeff Clune。2015。通过映射精英个体展现搜索空间。（2015年4月）。arXiv:1504.04909 [26] Olle Nilsson 和 Antoine Cully。2021。策略梯度辅助的 MAP-Elites。载于遗传与进化计算大会论文集。866–875。[27] Thomas Pierrot、Guillaume Richard、Karim Beguir 和 Antoine Cully。2022。多目标质量-多样性优化。载于遗传与进化计算大会论文集。139–147。[28] Justin K. Pugh、Lisa B. Soros 和 Kenneth O. Stanley。2016。质量-多样性：进化计算的新前沿。《机器人与人工智能前沿》3，7月（2016年7月），12。[29] Justin K. Pugh、L. B. Soros、Paul A. Szerlip 和 Kenneth O. Stanley。2015。直面质量-多样性的挑战。载于*GECCO 2015——2015*年遗传与进化计算大会论文集。计算机协会，纽约，美国纽约州，967–974。[30] Tim Salimans、Jonathan Ho、Xi Chen、Szymon Sidor 和 Ilya Sutskever。2017。进化策略：强化学习的一种可扩展替代方案。*arXiv* 预印本 *arXiv:1703.03864*（2017）。[31] Bryon Tjanaka、Matthew C Fontaine、Julian Togelius 和 Stefanos Nikolaidis。2022。近似梯度以实现强化学习中的可微质量-多样性。载于遗传与进化计算大会论文集。1102–1111。[32] Daan Wierstra、Tom Schaul、Tobias Glasmachers、Yi Sun、Jan Peters 和 Jürgen Schmidhuber。2014。自然进化策略。机器学习研究期刊15，1（2014），949–980。

数学论证

本节旨在提供数学证明，以论证本文所选方法的合理性。

引理1。 如果$(d_{(j)})$表示随机变量向量 $\boldsymbol{d}$ 的分量，则：

$$\sum_j Var\left[d^{(j)}\right] = E\left[\|\boldsymbol{d} - E\left[\boldsymbol{d}\right]\|_2^2\right]$$

证明。

$$\begin{aligned}\mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2\right] &= \mathrm{E}\left[\sum_j \left(d^{(j)} - \mathrm{E}\left[d^{(j)}\right]\right)^2\right] \\ &= \sum_j \mathrm{E}\left[\left(d^{(j)} - \mathrm{E}\left[d^{(j)}\right]\right)^2\right] \\ &= \sum_j \mathrm{Var}\left[d^{(j)}\right]\end{aligned}$$

□

定理1。 如果我们考虑一个有界描述符空间，以及一个满足 $E\left[\boldsymbol{d}\right] \in C$ 的单元格C，那么通过在问题$P2$中最大化概率 $P\left(\boldsymbol{d} \in C\right)$，我们可以最小化描述符方差之和 $\sum_j Var\left[d_{(j)}\right]$ 的一个上界，这些方差出现在问题$P1$中。

证明。利用引理1和全期望公式，我们得到：

$$\begin{aligned}\sum_j \mathrm{Var}\left[d^{(j)}\right] &= \mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2\right] && \text{see Lemma 1} \\ &= \mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2 \,|\boldsymbol{d} \in C\right]\mathrm{P}\left(\boldsymbol{d} \in C\right) \\ &\quad + \mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2 \,|\boldsymbol{d} \notin C\right]\mathrm{P}\left(\boldsymbol{d} \notin C\right) \\ &= \mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2 \,|\boldsymbol{d} \in C\right]\mathrm{P}\left(\boldsymbol{d} \in C\right) \\ &\quad + \mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2 \,|\boldsymbol{d} \notin C\right]\left(1 - \mathrm{P}\left(\boldsymbol{d} \in C\right)\right)\end{aligned}$$

此外，$\mathrm{P}\left(\boldsymbol{d} \in \mathrm{C}\right) \leq 1$。若将 dC 记为单元格C中两点之间的最大值，那么，由于$\mathrm{E}\left[\boldsymbol{d}\right] \in \mathrm{C}$，因此：

$$\forall \boldsymbol{d} \in C \;\; \|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2 \leq d_C^2$$

因此：

$$\mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2 \,|\boldsymbol{d} \in C\right] \leq d_C^2$$

此外，描述符空间有界，因此存在一个常数 α，使得

$$\mathrm{E}\left[\|\boldsymbol{d} - \mathrm{E}\left[\boldsymbol{d}\right]\|_2^2 \,|\boldsymbol{d} \notin C\right] \leq \alpha$$

最终，我们得到以下不等式：

$$\sum_j \mathrm{Var}\left[d^{(j)}\right] \leq d_C^2 + \alpha(1 - \mathrm{P}\left(\boldsymbol{d} \in C\right))$$

因此，在定理所述的假设下，通过最大化$\mathrm{P}\left(\boldsymbol{d} \in \mathrm{C}\right)$，我们可以最小化描述符方差之和的一个上界。

□

定理2。 如果适应度函数有界，那么最大化 $\widetilde{o}C$ 的期望值，也会最大化问题$P2$的两个目标的一个下界：$P\left(\boldsymbol{d} \in C\right)$ 和期望适应度 $E\left[f\right]$。

证明。根据全期望公式以及 $\widetilde{o}$ 的定义（见公式 1）：

$$\begin{aligned}\mathrm{E}\left[\widetilde{o}\right] &= \mathrm{E}\left[\widetilde{o}|\boldsymbol{d} \in C\right]\mathrm{P}\left(\boldsymbol{d} \in C\right) \\ &\quad + \mathrm{E}\left[\widetilde{o}|\boldsymbol{d} \notin C\right]\left(1 - \mathrm{P}\left(\boldsymbol{d} \in C\right)\right) \\ &= \mathrm{E}\left[f|\boldsymbol{d} \in C\right]\mathrm{P}\left(\boldsymbol{d} \in C\right) \\ &\quad + \mathrm{E}\left[f_{min} - \|\boldsymbol{d} - \boldsymbol{c}\|_2 \,|\boldsymbol{d} \notin C\right]\left(1 - \mathrm{P}\left(\boldsymbol{d} \in C\right)\right) \\ &\geq \mathrm{E}\left[f|\boldsymbol{d} \in C\right]\mathrm{P}\left(\boldsymbol{d} \in C\right) + f_{min}\left(1 - \mathrm{P}\left(\boldsymbol{d} \in C\right)\right) \\ &\geq \left(\mathrm{E}\left[f|\boldsymbol{d} \in C\right] - f_{min}\right)\mathrm{P}\left(\boldsymbol{d} \in C\right) + f_{min}\end{aligned}$$

因此：

$$\begin{aligned}\mathrm{P}\left(\boldsymbol{d} \in C\right) &\geq \frac{\mathrm{E}\left[\widetilde{o}\right] - f_{min}}{\mathrm{E}\left[f|\boldsymbol{d} \in C\right] - f_{min}} \\ &\geq \frac{\mathrm{E}\left[\widetilde{o}\right] - f_{min}}{f_{max} - f_{min}}\end{aligned}$$

这意味着，通过最大化 $\mathrm{E}\left[\widetilde{o}\right]$，我们就能最大化 $\mathrm{P}\left(\boldsymbol{d} \in \mathrm{C}\right)$ 的一个下界。

此外，如果对期望适应度应用全期望公式：

$$\begin{aligned}\mathrm{E}\left[f\right] &= \mathrm{E}\left[f|\boldsymbol{d} \in C\right]\mathrm{P}\left(\boldsymbol{d} \in C\right) \\ &\quad + \mathrm{E}\left[f|\boldsymbol{d} \notin C\right]\left(1 - \mathrm{P}\left(\boldsymbol{d} \in C\right)\right)\end{aligned}$$

特别是，我们知道：

$$\mathrm{E}\left[f|\boldsymbol{d} \notin C\right] \geq \mathrm{E}\left[f_{min} - \|\boldsymbol{d} - \boldsymbol{c}\|_2 \,|\boldsymbol{d} \notin C\right]$$

而且我们已在上文证明：

$$\begin{aligned}\mathrm{E}\left[\widetilde{o}\right] &= \mathrm{E}\left[f|\boldsymbol{d} \in C\right]\mathrm{P}\left(\boldsymbol{d} \in C\right) \\ &\quad + \mathrm{E}\left[f_{min} - \|\boldsymbol{d} - \boldsymbol{c}\|_2 \,|\boldsymbol{d} \notin C\right]\left(1 - \mathrm{P}\left(\boldsymbol{d} \in C\right)\right)\end{aligned}$$

因此：

$$\mathrm{E}\left[f\right] \geq \mathrm{E}\left[\widetilde{o}\right]$$

这意味着，通过最大化 $\mathrm{E}\left[\widetilde{o}\right]$，我们可以最大化期望适应度 $\mathrm{E}\left[f\right]$ 的一个下界。

□

B 档案

图 7 和图 8 分别展示了所有算法在机械臂任务和行走者任务中返回的档案可视化结果。这些图补充了图 6 中提供的蚂蚁任务可视化结果。

C 行走者校正档案分析

我们在图 9 中展示了校正档案和 ARIA- PGA 初始化档案的负描述符方差（NDV）得分及其概率值。我们注意到，ARIA-PGA 初始化获得的档案在不同单元格中损失了一些期望适应度。尽管如此，ARIA- PGA 初始化主要是在优化性能之前优先优化可复现性（见公式 1），而且那

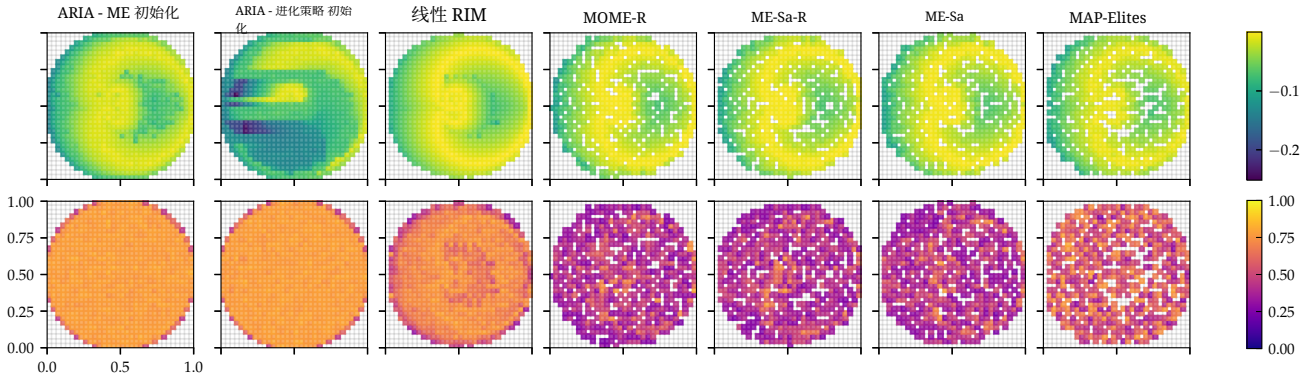

图 7：机械臂任务中各算法返回的档案内，期望适应度、NDV 分数以及属于单元格的概率分布。对于每种算法，所选档案是在 10 次重复实验中达到中位数 QD 分数的档案。由于机械臂任务并非 QD-RL 任务，因此无法直接将 PGA-ME 应用于该任务。此外，未展示方差图，因为在机械臂任务中，所有解的描述符方差均相同（见公式 3）。

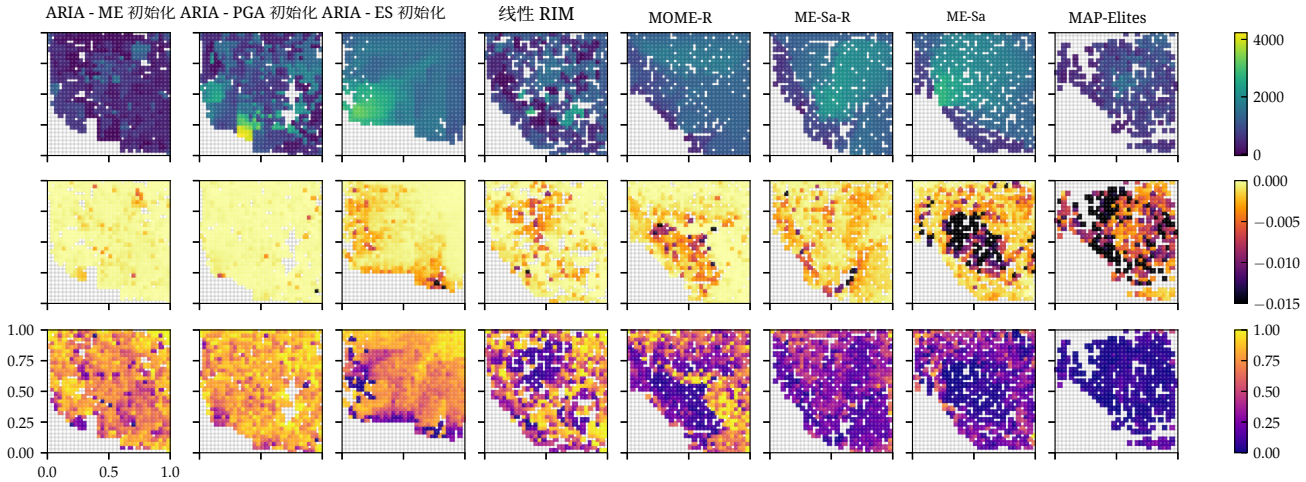

图 8：行走者任务中各算法返回的档案内，期望适应度、NDV 分数以及属于单元格的概率分布。对于每种算法，所选档案是在 10 次重复实验中达到中位数 QD 分数的档案。

图 9 中的 NDV 和概率值证实了这一点。有趣的是，ARIA-PGA 初始化档案上出现了一些空缺，位于校正档案中原本被占据的单元格；这意味着可复现性改进机制可能需要进行一些额外的微调。

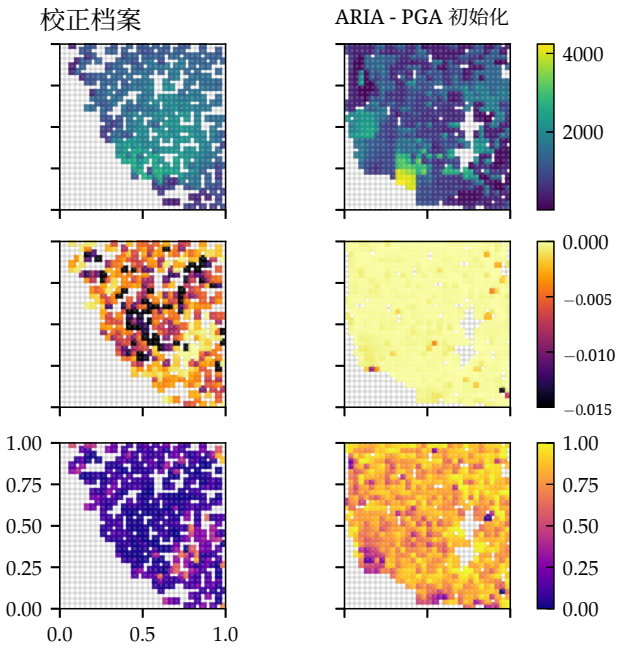

图9：图5中行走者任务的校正档案和ARIA-PGA 初始化档案的期望适应度、NDV 分数以及属于单元格的概率分布。