arXiv:2603.03932v1 [cs.NI] 4 Mar 2026

随机网络控制的离线强化学习算法选择

Nicolas Helson、Pegah Alizadeh、Anastasios Giovanidis爱立信研究院，*Massy*，法国邮箱：nicolas.helson@gmail.com, {firstname.lastname}@ericsson.com

摘要——**离线强化学习（RL）是下一代无线网络的一种前景广阔的方法。在这类网络中，在线探索并不安全，而且大量运营数据可以在模型生命周期的不同阶段重复利用。然而，对于无线系统因衰落、噪声和业务流量移动性而固有的真实随机动态，离线强化学习算法的表现仍缺乏充分认识。为弥补这一空白，我们在一个开放接入的随机电信环境（mobile-env）中评估了基于贝尔曼的方法（保守Q学习）、基于序列的方法（决策 Transformer），以及混合方法（评论家引导的决策 Transformer）。结果表明，保守Q学习在不同随机性来源下始终能生成鲁棒性更强的策略，因此是生命周期驱动型 AI 管理框架中可靠的默认选择。若有足够多的高回报轨迹，基于序列的方法仍具有竞争力，并且可能优于基于贝尔曼的方法。这些发现为 AI 驱动的网络控制流程（如 O-RAN 和未来的 6G 功能）中的离线强化学习算法选择提供了实用指导；在这类场景中，鲁棒性和数据可用性是关键的运营约束。**

关键词——**离线强化学习、网络管理、自治网络、人工智能生命周期管理、决策 Transformer、随机网络环境、O-RAN**

I. 引言

新一代无线网络采用零接触能力，融入全面自动化，尤其是在参数调优方面；基于学习的控制功能正越来越多地被部署，并作为受管 AI 生命周期的一部分进行更新（例如 O-RAN 和未来的 6G 架构）。强化学习（RL）是实现预期适应性和韧性的关键。基于价值的方法和策略梯度方法能够成功解决高维决策任务 [1]，[2]。然而，在线强化学习 [3]面临挑战，因为它需要与真实环境或模拟环境直接交互，才能充分探索各种可用调优选项的影响。在真实环境中进行探索可能代价高昂且不安全，因为次优配置可能会降低网络性能并中断所提供的服务。因此，实践中通常在模拟器上进行训练，这些模拟器以不同抽象层次模拟现实。然而，即使在模拟器上，探索也可能很慢，因为评估特定参数调优对各项网络关键性能指标（KPI）的影响需要时间。此外，训练好的策略在部署后还需要进一步微调，以利用真实数据进行调优和探索，从而适应真实环境的具体情况。

另一方面，无线网络运营商能够从部署在城市和农村地区的网络中收集大量数据。这些数据通常是定期收集的汇总 KPI，或是在网络配置发生变化时收集的汇总 KPI。这些数据集以多维时间序列的形式呈现，描述状态（观测 KPI ）、动作（参数调优）和奖励（目标 KPI ）的序列。这类结构化轨迹自然符合马尔可夫决策过程（MDP）框架，因此适合用于训练强化学习智能体。为此，离线强化学习 [4]应运而生，成为一种自然选择，因为它能够利用这些预先收集的数据集来推导出高性能的新策略，而无需进行显式探索。这一观点与 O-RAN联盟定义的人工智能/机器学习生命周期相一致，在该生命周期中，数据收集、数据准备和模型训练被明确划分为网络控制功能人工智能/机器学习工作流中的不同阶段 [5]。

自然，离线强化学习也带来了新的挑战，直接影响算法选择和部署决策。由于依赖通过某种行为策略收集的固定数据集，它可能会导致对罕见或从未观测到的动作的价值估计不准确 [4],，这被称为动作分布外（OOD）问题。此外，为了使任何离线强化学习方法都能学到有意义的策略，数据集应包含足够多转移到高回报区域的状态转移。

处理这些问题有两种重要方法：一种基于传统的基于贝尔曼方程的形式，另一种采用序列方法；两者在鲁棒性、数据需求和运行适用性方面各有不同的权衡。保守 *Q* 学习（CQL） [6] 是一种基于贝尔曼方程的方法，它通过对价值函数进行正则化来减少过高估计，从而解决其中一些问题。近来，决策*Transformer*（DT） [7] 及其变体（如评论家引导的 DT） [8] 已成为一种很有前景的替代方案。DT 将强化学习构建为条件序列建模问题，绕过显式自举，转而根据期望的未来累积回报（即回报目标）预测动作，并借助大规模 Transformer 模型的进展。

电信环境固有的随机性给离线强化学习带来了另一项重大挑战，直接影响在线网络中自主决策的可靠性。这种随机性可能体现在马尔可夫决策过程的不同组成部分中，例如初始状态分布、状态/动-

作状态转移或奖励函数，其强度和结构各不相同。这被称为偶然不确定性，即[9], 环境固有且无法降低的不确定性。另一种不确定性，即认识不确定性，源于模型或训练数据的局限性，原则上可以通过增加数据或改进建模来降低。

贡献：尽管已有研究将决策 Transformer（DT）与 CQL 等基于贝尔曼的方法进行比较 [10], [11], ，但这些研究要么忽略了随机性 [10] ，要么在确定性数据上完成训练后才评估随机性 [11]。本研究的一项重要首要贡献，是在自然融入随机性的环境中比较这些方法。我们在一个面向电信领域的框架 mobile-env 上开展实验 [12], ，该框架支持可控的随机性，能够代表真实世界的无线多小区场景。研究的两种随机性分别是：业务流量移动性和信道衰落。

最重要的是，本研究将 CQL 与 DT 的比较扩展到不同类型和不同程度的偶然不确定性，并纳入已被证明优于标准决策变换器的序列方法。具体而言，我们评估了一种融入评论家目标的 DT 变体——评论家引导 DT [8]。为求全面，我们还开展了数据集消融研究，以评估高随机性条件下认知不确定性的影响。

具体而言，我们的核心问题是：**随机性如何影响离线强化学习算法在无线通信中的性能与部署？**为回答这一问题，我们提出并解答一系列实用子问题，以评估基于贝尔曼的与基于序列的离线强化学习方法的行为：

- 用户移动性（状态转移随机性）如何扰动电信智能体的性能？（第 V-A 节）
- 在用户移动性条件下，数据集质量如何影响电信智能体的性能？（第 V-B 节）
- 信道衰落（奖励随机性）如何扰动电信智能体的性能？（第 V-C 节）

II. 相关工作

离线强化学习正越来越多地应用于无线通信系统。近期工作 [13] 研究了数据集构成对无线系统离线强化学习性能的影响。 [14] 的研究比较了在线与离线强化学习方法（包括决策 Transformer 和 CQL）在下行链路自适应问题上的表现。类似地， [15] 探讨了将离线强化学习——尤其是 CQL——用于 RAN 切片优化。 [16] 中的工作利用决策 Transformer 和 CQL 学习最优切换参数调优。然而，这些工作均未明确将随机性视为电信离线强化学习中的一项主要挑战。

决策 Transformer（DT） [7] 是离线强化学习中一类重要的序列方法。然而，DT 有两个主要局限：拼接能力较弱（即难以将次优轨迹片段组合成更优的轨迹） [8], [17], ，并且对回报条件化引入的随机性敏感 [18]。当高回报源于运气，而非持续的最优行为时，它可能会失效。为解决 DT 的局限，人们提出了若干方法，通常会引入依据贝尔曼方程原理训练的评论家。Q学习决策变换器 [17] 采用 CQL 风格的回报重标记，以减少过高估计偏差；评论家引导的 DT（CGDT） [8]则预训练评论家来指导策略学习，从而增强对次优轨迹的拼接能力，并减少对“幸运”高回报样本的依赖。ES-PER [18]通过对轨迹数据进行聚类来增强对随机性的鲁棒性，而 ACT [19]则引入优势条件化，以稳定训练。

此外，还有其他基于贝尔曼的方法可用于高效的离线强化学习。与前文介绍的保守 Q 学习（CQL） [6],类似，隐式 Q 学习（IQL） [20]通过采用分位数回归目标，将价值更新限制在分布内动作上，从而缓解动作分布外问题 [4] 。

已有研究通过在测试时引入噪声，考察了随机性对离线强化学习性能的影响 [11]。其他研究 [18],[19] 仅对随机性设置下的 CQL 和决策 Transformer（DT）进行了有限比较。文献 [18] 研究了状态转移随机性，结果表明，在低随机性的四子棋环境中，CQL 优于 DT；而在 2048 游戏中，DT 则优于 CQL。ACT [19] 在 D4RL 基准中对选定动作引入噪声，发现 CQL 的表现始终优于 DT。

III. 已研究的算法

强化学习（RL）以马尔可夫决策过程（MDP）为基础，该过程由表示环境的状态、影响状态转移的动作，以及用于衡量结果优劣的奖励构成。状态／动作转移概率和奖励函数通常未知或难以求解，因此需要采用数据驱动的无模型强化学习。值得注意的是，马尔可夫决策过程（MDP）可能呈现不同形式的随机性。最常见的来源是状态转移随机性，即在给定状态下采取某个动作时，可能根据某一概率分布转移到多个不同的后续状态。另一种来源是奖励随机性，即接收的奖励本身是一个以状态和动作为条件的随机变量。虽然在受测环境中不太常见，但奖励随机性实际上自然存在于电信场景中，主要由无线信道衰落导致；因此，接收比特率（奖励）并非确定值。其他随机性的来源还包括初始状态随机性，因为环境通常以随机状态初始化；但在电信环境中，由于网络持续、不中断地运行，这一因素的影响较小。

强化学习的目标是学习一种策略，将每个状态映射为一个动作，从而最大化预期未来累积奖励，这称为回报。序列方法中常用的一个相关概念是剩余回报，其定义为从当前时间步 t 到

轨迹 T结束时的奖励总和：$\sum_{i=t}^{T} r_i$。接下来，我们将更详细地介绍所研究的算法。

CQL [6]。这遵循经典的强化学习范式：策略由学习得到的价值函数推导而来，该函数用于估计每个状态–动作对的预期累积奖励。为了解决价值学习中的动作分布外问题，作者在标准贝尔曼误差目标中加入了保守惩罚项。该惩罚项抑制对未见动作赋予高价值，同时使分布内动作的价值保持稳定。

决策 Transformer [7]。与此同时，决策 Transformer 提出了一系列序列建模方法，这些方法不学习价值函数，而是将强化学习表述为根据过去的轨迹预测下一动作。与仅依赖当前状态的标准强化学习策略不同，决策 Transformer 还以先前状态、先前动作以及未来信息为条件。该策略通过对 Transformer 进行监督训练来学习；Transformer 能够关注相关的过往元素和关键的未来信号：剩余回报。在推理时，模型会收到一个期望回报目标，并尝试生成能够实现该目标的动作。因此，决策 Transformer 能够遵循不同的目标回报，而基于贝尔曼的方法则侧重于学习单一的最优策略。为了使模型能够表示数据集中多样化的行为，训练和推理期间均采用上下文长度 K，以限制输入模型的过往信息量。较长的上下文有助于模仿数据集中的轨迹，而较短的上下文则能带来更灵活的动作预测。

CGDT [8]。学习遵循回报目标可能会受到限制，尤其是在数据集中包含次优轨迹时——这些轨迹中有良好的动作，却只能获得较低回报；或者在随机环境中，高回报可能只是运气使然。为解决这一问题，CGDT首先以监督方式学习一个评论家，根据过去的轨迹上下文预测给定动作的预期回报。随后修改DT目标函数，使其在两个部分之间取得平衡：一是以过去的状态和目标回报为条件的标准DT监督动作预测损失，二是额外的评论家引导项，鼓励模型选择预测表现优于数据集回报的动作。这样一来，模型便能超越模仿，力争获得高于数据中所观察到的回报。

IV. 电信环境

我们考虑一种基于 mobile-env [12]，的电信环境。mobile-env 是一款开放、轻量级的蜂窝网络模拟器，旨在支持用户移动性和动态用户–基站关联，如图1所示。我们对原始[1] 环境进行了改造，将其随用户数量呈指数级增长的动作空间替换为更现实的方案：采用关联阈值，并为每个基站仅设置三种可能的动作。这一改动符合现实中的参数调优需求，同时使优化问题变得易于处理。本文介绍改造后的动作空间版本。

[1]源代码：https://github.com/stefanbschneider/mobile-env/tree/main

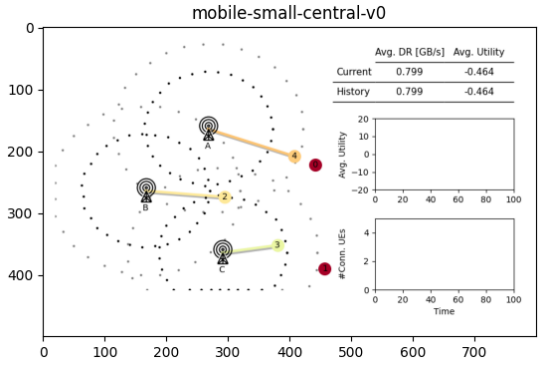


图1. mobile-env环境示意图。

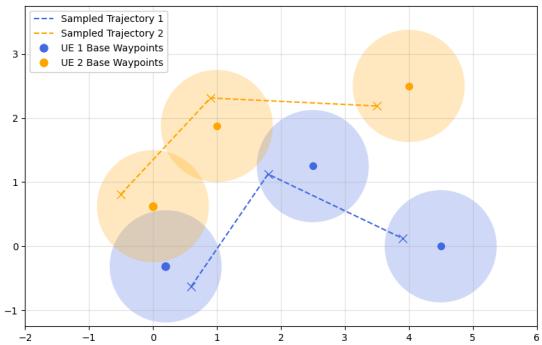


图2. 随机航点模型的低移动性变体。

mobile-env 环境由多个基站（BS）和用户设备（UE）组成，每个用户设备都连接到一个或多个基站。用户具有随机的移动轨迹，这使问题更具挑战性。目标是最大化平均用户效用，该效用定义为用户从所连接基站接收的数据速率的函数。直观地说，同时将用户连接到多个基站可以提高其数据速率，但在已连接的用户之间共享资源会降低所有用户的数据速率，从而形成负载–性能权衡。

考虑一个基站及与其连接的用户。用户通过该基站获得的数据速率取决于信噪比（SNR），即用户所感受到的信号强度，以及基站的调度策略；该策略决定如何在已连接用户之间共享资源。在本研究中，我们采用RateFair调度策略，为所有已连接用户分配相同的数据速率。这意味着，与附近用户相比，距离较远的用户需要更多资源才能达到相同的数据速率。最后，将用户通过其所连接的所有基站获得的数据速率相加，得到该用户的最终数据速率。

数据速率取决于用户与基站的连接。在我们使用的 mobile-env 版本中，每个基站都有一个 SNR 阈值，用于控制哪些用户可以连接，从而通过调高或调低该阈值来进行负载调节。简单来说，当用户的信噪比超过某个基站的阈值时，该用户就会连接到该基站。如果阈值设置得太高，就没有用户能够连接，奖励为 0。如果阈值设置得太低，所有用户都会连接到所有基站，而由于资源共享，奖励也会再次降低。

现在我们来介绍 mobile-env 的强化学习建模。我们设置 5 个用户设备和 3 个基站，以研究一个规模较小但具有足够随机性且有意义的场景。

- 状态空间：状态包括所有基站（BS）的当前连接阈值、每个基站—用户对的当前信噪比值，以及用户先前的效用。因此，状态维度为 $|B_s|+|U_e||B_s|+|U_e| = 23$，其中 $|U_e|$ 和 $|B_s|$ 分别表示系统中的用户设备和基站数量。
- 动作空间：动作空间为离散型。对于每个基站，动作包括略微提高、略微降低或保持当前阈值不变。实际应用中，更新步长设为信噪比总范围的十分之一。因此，动作空间有 $|B_s|$ 个维度，大小为$3^{|B_s|} = 9$。

•奖励：奖励定义为所有用户设备的平均效用。如上所述，效用随比特率单调递增；比特率取决于信噪比、用户—基站关联以及基站分配的资源。奖励可能因信道衰落而具有随机性（见下文）。更多详情请参阅附录。

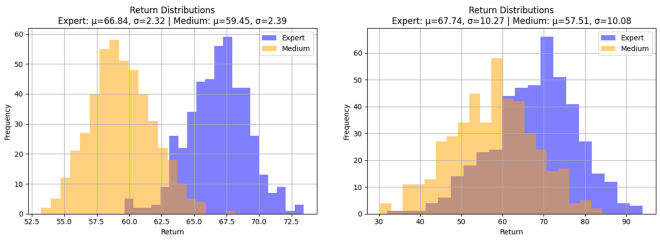
在本研究中，我们考察了两种随机性来源：（i）用户移动性，它会使状态转移函数具有随机性；（ii）信道衰落，它会使奖励函数具有随机性。

用户移动性： 遵循随机航点（RWP）模型 [21]。每个用户以固定速度朝随机采样的航点移动，到达先前的航点后，再抽取新的航点。由于用户移动会影响未来的信噪比值，因此会导致状态转移具有随机性。为了研究如何降低转移随机性，我们还定义了一种低移动性 变体，并将其纳入环境。在这种变体中，每个航点都在采样位置周围的受限区域内抽取，从而降低轨迹的变化程度。图2展示了这一机制。

信道衰落： 将其建模为对用户信噪比的随机扰动，以反映障碍物和多径传播等因素的影响。尽管衰落在蜂窝通信中十分常见，原始代码并未将其纳入，因此我们添加了这一因素。我们重点研究瑞利衰落，它代表不存在视距分量的高度随机环境。衰落只会引入奖励随机性：在每个时隙，都会为每个用户设备–基站对抽取一个衰落系数，并在奖励计算中将信噪比乘以该系数的平方。状态信噪比不受扰动，这反映了实际系统中使用平均测量值的情况。因此，衰落会影响奖励，但不会改变状态转移。

V. 实验

离线强化学习的第一步是收集离线数据集；在实际应用中，我们通常假设该数据集已可用。按照文献中针对 Gym 环境和 D4rl 数据集的常见做法 [22]，，我们通过训练在线强化学习智能体生成这些数据集。对于 mobile-env 的离散动作设置，我们使用了 d3rlpy [23] 中的 Double DQN 算法。该流程得到了一个专家策略，并且



(a) 有限移动随机性。　　(b) 高移动性随机性。

图3. mobile-env在不同移动性随机性设置下的训练轨迹回报分布。

通过保存训练早期阶段的模型参数，我们获得了一个表现中等的策略。随后，我们构建了一个中等策略-专家数据集，其中包含使用专家策略采样的轨迹和使用中等策略采样的轨迹。在 mobile-env 中，我们构建了一个数据集，其中包括 500 条专家轨迹和 500 条中等水平轨迹，每条轨迹的时域长度为 100，总计 100k 步。我们通过在线运行 30 条轨迹来评估训练好的智能体，并报告回报均值和标准差。

为便于阅读，我们按照标准离线强化学习实践 [22]，以专家的平均性能为上界、随机策略的平均性能为下界，将分数重新缩放到范围 $[0, 100]$,。重新缩放还便于在不同设置（例如随机性水平）之间进行比较。

A. 用户移动性（状态转移随机性）如何扰动电信智能体的性能？

我们现在分析 mobile-env 中状态转移随机性的影响。使用上述中等和专家数据集及评估方法，我们在两种移动性条件下训练所有算法：有限移动性和高移动性。图 3 展示了训练数据集的回报分布。在高移动性设置下，尽管平均回报差距相似，专家与中等策略的回报仍有更多重叠，这意味着中等策略偶尔可能凭运气胜过专家。这加剧了幸运回报问题，使随机性设置尤其具有挑战性，特别是对序列方法而言。

表 I 移动性随机性有限和高的情况下的重缩放平均回报（括号内为标准差），适用于 mobile−env。

模型	有限随机性	高随机性	Δ（高移动性随机性–有限随机性）
DT	95.1 (9.71)	81.5 (51.6)	-13.6
CGDT	**95.6 (9.75)**	83.0 (51.2)	-12.6
CQL	94.6 (13.1)	**84.8 (46.9)**	$\mathbf{-9.8}$

在分析完数据集后，我们现在转向算法结果（表 I）。在有限移动性条件下，所有方法的表现都相近，这符合低随机性状态下的预期。CGDT（$L = 5$, $\tau_c = 0.5$, $\tau_p = 0.8$, $\alpha = 0.1$）相比决策 Transformer（$L = 5$）取得了小幅但令人鼓舞的提升，而 CQL（$\alpha = 1.0$，比率 $\ell_r = 3 \cdot 10^{-4}$）则表现出较高的方差。一种解释是，有限移动性限制了状态–动作

的多样性，使 CQL 更难在不依赖分布外动作的情况下学到可靠的价值函数。

在高移动性场景中，所有方法再次达到相近的平均性能，但由于移动模式多样，方差更大。此处 CQL（$\alpha = 0.1$，速率 $\ell_r = 3 \cdot 10^{-4}$）优于序列模型，这与其在随机环境中基于价值的鲁棒性相一致。移动性的增加也暴露了更多的状态-动作空间，从而改善了 CQL 的价值估计并缓解了分布外问题。CGDT（$L = 10$、 $\tau_c = 0.5$、 $\tau_p = 0.8$、 $\alpha = 0.2$）继续相对决策 Transformer（$L = 10$）表现出虽小但稳定的增益。考虑到有限移动性与高移动性之间的性能变化，所有方法均出现大幅下降：决策 Transformer 下降 13.6，CGDT 下降 12.6，CQL 下降 9.8。这证实了移动性引起的随机性会对离线强化学习性能产生不利影响，同时也凸显了 CQL 相对于序列方法具有更强的鲁棒性。

另一个值得注意的观察是，与 CQL 相比，序列方法，尤其是决策 Transformer，仍能取得出色的表现。这与文献中常见的报告结果形成对比 [11], [10]。通过处理更长的输入序列，序列模型能够利用过去的状态（信噪比）趋势来预测未来值；这在直线移动情况下尤其有用，使其优于仅依赖当前状态（信噪比）的基于贝尔曼的方法。数据集也可能包含足够多的专家轨迹，从而限制含噪次优轨迹的影响。为考察这些因素并评估认知不确定性，我们通过逐步移除中等水平数据和专家数据开展消融研究。

关键结果 1：用户移动性会对智能体产生负面影响，因为更高的状态转移随机性会导致所有方法的性能下降。不过，CQL 仍然最为稳健，其绝对性能最高，性能下降幅度也最小。

B. 数据集质量如何影响用户移动性下电信智能体的性能？（认知不确定性）

为研究在 mobile-env 的高移动性随机性下，数据数量和质量如何影响离线强化学习算法的性能，我们通过逐步从训练集中移除部分数据开展了消融研究。因此，我们在高偶然不确定性下提高了认知不确定性。我们采用本节开头最初收集的数据集和方法，并评估了三种变体：移除 50% 的专家数据（*–50%* 专家数据）、 50% 的中等水平数据（*–50%* 中等水平数据），以及两者各移除 50%（*–50%* 两者）。结果见表 II。结果以未缩减的完整数据集的专家均值进行缩放。 Δ 数值表示在完整数据集上训练所得的缩放结果与对应结果之间的性能差异。

首先，CQL 始终表现稳定：它在所有设置下均表现出色，展现了在各种认知不确定性设置中对随机性的鲁棒性，尤其是在专家轨迹有限时。序列方法的鲁棒性较差。当专家数据减少时（*–50%* 专家

表 II **高移动性**随机性下的消融研究。

设置	模型	平均回报（标准差）	Δ 与完整数据集相比
完整数据集	DT	81.5 (51.6)	-
	CGDT	83.0 (51.2)	-
	CQL	**84.8 (46.9)**	-
-50% 中等/专家	DT	73.2 (53.3)	-8.3
	CGDT	78.1 (55.6)	-4.9
	CQL	**81.0 (51.2)**	$\mathbf{-3.8}$
专家的 -50%	DT	59.9 (52.8)	-21.6
	CGDT	67.9 (52.6)	-15.1
	CQL	**81.4 (48.3)**	$\mathbf{-3.40}$
中等的 -50%	DT	81.7 (50.5)	$+0.20$
	CGDT	**83.5 (53.4)**	$\mathbf{+0.50}$
	CQL	82.3 (52.5)	-2.50

以及*–50%* 两者），它们表现不佳；但仅移除中等水平数据时，表现良好。CGDT 在所有设置下始终优于决策 Transformer。总体而言，CQL 的表现最佳，但*–50%* 中等水平的情况除外，此时 CGDT 略胜一筹。

事实上，相较于数据集构成（即专家轨迹与中等水平轨迹之间的平衡），CQL 对数据集规模更为敏感。在使用完整数据集时，其性能达到最高；而在两者均减少 *50%*时最低。相比之下，中等水平数据减少 *50%*和专家数据减少 *50%*都会导致性能适度下降。

对于序列方法而言，数据质量似乎比数据数量更为重要。有趣的是，移除 50% 的中等水平数据并不会损害决策 Transformer 或 CGDT 的性能，反而会使其略有提升。此外，当专家数据和中等水平数据同时减少时（两者均减少 *50%*），这些方法的表现优于仅减少专家数据时（专家数据减少 *50%*）。专家样本过少可能会增加回报条件过拟合于“幸运的”中等水平轨迹的风险。另一方面，拥有足够比例的专家数据能够稳定学习过程。这与先前的工作 [18] 一致，后者表明序列方法可能会受到高回报但不具代表性的轨迹误导，并提出了一种切实可行的缓解措施：纳入足够的专家数据，以限制这种偏差。

关键结果 2： 在高移动性随机环境中，除中等水平数据减少时由 CGDT 领先外，CQL 在所有认知不确定性配置下均优于序列方法。CQL 对数据数量更敏感，而序列方法受数据质量的影响更大。

C. 信道衰落（奖励随机性）如何影响电信智能体的性能？

为了考察奖励层面的随机性所带来的影响，我们接下来在高移动性场景的基础上研究瑞利信道衰落，因为单独的移动性并未显示算法之间存在显著的性能差异，而且移动性本身就是 mobile-env 框架的固有组成部分。与前文一样，我们采用本节开头所述的数据集采样、评估流程和缩放方法。

表 III 报告了瑞利衰落下离线强化学习算法的缩放后性能，以及它们相对于无衰落情况的差-

表 III 瑞利衰落高移动性下的性能。

设置	模型	平均回报（标准差）	Δ 与无衰减相比
瑞利	DT	47.71（65.57）	-33.8
	CGDT	80.38 (62.14)	-2.62
	CQL	**93.97 (64.35)**	**+9.17**

异。缩放时使用高移动性和瑞利衰落数据集中的专家平均性能。 Δ 数值表示与无衰落情况下的缩放后数值（使用高移动性数据集中的专家均值）之间的差距。CQL 仍然是最稳定的方法，取得了最高的平均回报和最低的方差，这表明它对奖励随机性相对不敏感，也证实了基于价值的学习在奖励波动时具有鲁棒性。序列方法的表现则有所不同：决策 Transformer 表现不佳，可能是因为强烈的奖励随机性掩盖了优劣动作之间的区别；而 CGDT 相比决策 Transformer 有显著提升，但仍不及 CQL。

与无衰落情况相比的差值提供了更多见解。面对状态转移随机性和奖励随机性，CQL 均能保持其性能。相比之下，DT 的性能较无衰落结果大幅下降，这表明 DT 能够容忍中等程度的随机性，但当不确定性增加且来源多样时，其性能便会崩溃。然而，CGDT 表现出强得多的鲁棒性，性能仅略有下降。

关键结果3： 在加入瑞利衰落后，CQL 仍是表现最稳定的方法，而 CGDT 相比 DT 有显著提升，其归一化性能与无衰落情况相近。

VI. 结论

在本研究中，我们在一个逼真的电信环境（ mobile-env）中，比较了不同状态转移随机性和奖励随机性水平下的离线强化学习算法。我们评估了基于贝尔曼方程的方法（CQL）和序列方法（DT 及其评论家增强变体 CGDT），重点考察用户移动性和信道衰落所引发的随机性。结果表明，CQL 整体上最为稳健，而序列方法仍具有竞争力。从系统运行和生命周期视角来看，在高度随机的场景中应优先选择 CQL，尤其是在存在多个随机性来源且有充足数据可用时。序列方法，特别是相较于 DT 始终表现更优的 CGDT，更适用于随机性较弱、数据集较小且包含高回报轨迹的场景；随着系统运行过程中数据质量不断提升，这些方法也可能变得更具吸引力。

参考文献

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski 等，“通过深度强化学习实现人类水平的控制”，《自然》，第 518 卷，第 7540 期，第 529–533 页，2015年。

[2] T. Haarnoja、A. Zhou、P. Abbeel 和 S. Levine，《软演员-评论家：采用随机演员的离线最大熵深度强化学习》，载于《国际机器学习会议》。PMLR，2018年，第 1861–1870 页。[3] K. Yang、C. Shen 和 T. Liu，《基于深度强化学习的无线网络优化：一项比较研究》，载于《IEEE INFOCOM 2020——IEEE计算机通信会议研讨会》（INFOCOM WKSHPS），2020年，第 1248–1253 页。[4] S. Levine、A. Kumar、G. Tucker 和 J. Fu，《离线强化学习：教程、综述及开放问题展望》，arXiv 预印本 arXiv:2005.01643，2020年。[5] 《O-RAN工作组2人工智能/机器学习工作流程说明与要求》，O-RAN联盟，技术报告 O-RAN.WG2.AIML-v01.03，2023年。[6] A. Kumar、A. Zhou、G. Tucker 和 S. Levine，《面向离线强化学习的保守Q学习》，《神经信息处理系统进展》，第 33 卷，第 1179–1191 页，2020年。[7] L. Chen、K. Lu、A. Rajeswaran、K. Lee、A. Grover、M. Laskin、P. Abbeel、A. Srinivas 和 I. Mordatch，《决策 Transformer：通过序列建模实现强化学习》，《神经信息处理系统进展》，第 34 卷，2021年。[8] Y. Wang、C. Yang、Y. Wen、Y. Liu 和 Y. Qiao，《评论家引导的决策变换器：面向离线强化学习》，载于《AAAI人工智能会议论文集》，第 38 卷，第 14 期，2024年，第 15 706–15 714 页。[9] O. Lockwood 和 M. Si，《深度强化学习中的不确定性综述》，2022 [ ] 2208 09052 。在线。获取地址：https://arxiv.org/abs/。[10] Y. Omori、Z. Dong 和 K. Ross，《离线强化学习中是否应该优先选择决策 Transformer? 》，arXiv 预印本 arXiv:2507.10174，2025年。[11] P. Bhargava、R. Chitnis、A. Geramifard、S. Sodhani 和 A. Zhang，《何时应优先选择用于离线强化学习的决策 Transformer? 》，《国际学习表征会议》（ICLR），2024年。[12] S. Schneider、S. Werner、R. Khalili、A. Hecker 和 H. Karl，《 mobile-env：面向无线移动网络强化学习的开放平台》，载于 NOMS 2022——2022年 IEEE/IFIP网络运维与管理研讨会，2022年。[13] K. Yang、C. Shi、C. Shen、J. Yang、S.-P. Yeh 和 J. J. Sydir，《基于混合数据集的无线网络优化离线强化学习》，《IEEE无线通信汇刊》，第 23 卷，第 10 期，2024年，第 12 703–12 716 页。[14] S. Peri、A. Russo、G. Fodor 和 P. Soldati，《用于下行链路自适应的离线强化学习与序列建模》，载于《2025年IEEE通信与网络机器学习国际会议》（ICMLCN），2025年。[15] K. Yang、S. ping Yeh、M. Zhang、J. Sydir、J. Yang 和 C. Shen，《利用离线强化学习推进无线接入网切片》，2023年。[16] [ ] 2312 10547 在线。获取地址：https://arxiv.org/abs/。P. Alizadeh、A. Giovanidis、P. Ramachandra、V. Koutsoukis 和 O. Arouk，《用于移动性鲁棒性优化的离线强化学习》，arXiv 预印本 arXiv:2506.22793，2025年。[17] T. Yamagata、A. Khalil 和 R. Santos-Rodriguez，《 Q 学习决策 Transformer：利用动态规划进行离线强化学习中的条件序列建模》，载于《国际机器学习会议》。PMLR，2023年，第 38 989–39 007 页。[18] K. Paster、S. McIlraith 和 J. Ba，《不能指望运气：决策 Transformer 和回报条件变换器为何在随机环境中失效》，《神经信息处理系统进展》，第 35 卷，38 979，2022年 38 966 页。–。[19] C.-X. Gao、C. Wu、M. Cao、R. Kong、Z. Zhang 和 Y. Yu，《ACT：通过优势条件化，借助动态规划赋能决策 Transformer》，《AAAI人工智能会议论文集》，第 38 卷，第 11 期，2024年 3 月，第 12 127–12 135 页。[20] I. Kostrikov、A. Nair 和 S. Levine，《采用隐式Q学习的离线强化学习》，arXiv 预印本 arXiv:2110.06169，2021年。[21] C. Bettstetter、H. Hartenstein 和 X. Pérez-Costa，《随机航点移动模型的随机特性》，《无线网络》，第 10 卷，第 5 期，2004年，第 555–567 页。[22] J. Fu、A. Kumar、O. Nachum、G. Tucker 和 S. Levine，《D4RL：深度数据驱动强化学习数据集》，arXiv 预印本 arXiv:2004.07219，2020年。[23] T. Seno 和 M. Imai，《d3rlpy：离线深度强化学习库》，《机器学习研究期刊》，第 23 卷，第 315 期，2022年，第 1–20 页。

附录

VII. 月球着陆器的其他实验

A. 环境与随机性设置

为了初步了解离线强化学习中与随机性相关的挑战，我们首先在著名的 LunarLander 环境中进行了实验[2]，然后再转向电信专用 mobile-env。下面将介绍这项工作。目标是让航天器在地面上降落于指定的目标区域。主要挑战在于稳定航天器，并逐步接近着陆垫，以实现安全着陆。引入外部风后，问题便具有随机性，因为外部风会扰动航天器的动力学并影响状态转移。

无风的 LunarLander 环境定义如下：

- **状态空间：**状态向量包括航天器的位置、速度、角位置、角速度，以及两个布尔变量，用于指示左腿或右腿是否与地面接触。每个初始状态均随机采样：速度随机化，而位置、角度和角速度保持固定。

- **动作空间：**动作向量包含两个连续分量（a_0，a_1），其中 $a_0 \in [-1,1]$ 控制主引擎（垂直推力），$a_1 \in [-1,1]$ 控制侧向引擎。

- **奖励函数：**该奖励会鼓励航天器向着陆垫靠近，并用支腿接触地面。成功着陆可获得 $+100$ 的奖励，而速度过快、倾斜角过大以及发生碰撞则会受到惩罚（-100）。

转移随机性（风）：航天器会受到如下形式的随机风信号扰动

$$w(t) = \tanh\big(\sin(2k(t+C)) + \sin(\pi k(t+C))\big),$$

其中，t 表示时间（序列）索引，$k = 0.01$、$C \sim$服从 Uniform$(-9999,\ 9999)$，且 $w(t) \in [-1,1]$。对于每条轨迹，都会对 C 进行两次独立采样，以生成 $w_F(t)$ 和 $w_T(t)$，分别定义作用于航天器的侧向风力和扭矩。这些信号乘以 $\alpha \in [0,2]$ 进行缩放，以控制风力强度；我们采用较强的风，令 $\alpha = 1.5$。总之，航天器受到侧向力 $\alpha w_F(t)$ 和扭矩 $\alpha w_T(t)$ 的扰动，从而在动力学中引入状态转移随机性。

B. 实验

对于LunarLander，我们的目标是评估随机性是否确实会影响离线强化学习算法，尤其是风引起的状态转移随机性。由于此处的动作空间是连续的，我们使用软演员-评论家[3] 在线获得了一个专家策略。为评估随机性的影响，我们在有风和无风两种情况下，对所研究的三种离线强化学习算法进行了训练和评估，环境均为LunarLander。我们使用中等策略-专家数据集进行了训练和评估，所得性能指标见表 IV。

表 IV 月球着陆器：重缩放平均回报（标准差）。

模型	无随机性	高随机性	Δ (高–无)
DT	90.4 (16.2)	83.5 (25.5)	-6.9
CGDT	91.7 (14.5)	91.6 (19.6)	$\mathbf{-0.1}$
CQL	**101.2 (14.2)**	**96.9 (17.9)**	-4.3

这些结果表明，在确定性和随机性设置下，CQL 在平均回报和标准差方面均优于序列方法。值得注意的是，在高随机性场景中，CQL 相较于 DT 和 CGDT 仍保持明显优势，而所有方法都难以达到专家级性能（100）。

更有意思的是，确定性设置与随机性设置之间的性能落差，凸显了各算法应对随机扰动的鲁棒性。CGDT 受影响最小（$\Delta = -0.1$），这表明与 DT 相比，评论家组件增强了其对状态转移随机性的鲁棒性。这也有助于 CGDT 在两种设置下缩小与 CQL 的性能差距。相反，DT 的性能显著下降（$\Delta = -6.9$）；而 CQL 尽管整体表现出色，也出现了明显退化（$\Delta = -4.3$）。

总而言之，随机性确实会影响离线强化学习方法：它会显著降低 DT 和 CQL 的性能，而 CGDT 在确定性和随机性设置下均保持稳定。尽管如此，CQL 仍取得了最高的整体回报。

VIII. 修改后的 环境 MOBILE-ENV

A. 有限移动性

在实际应用中，mobile-env 定义了一个 200×200 正方形地图。对于有限随机性设置，初始位置在半径为 20（地图尺寸的十分之一）的圆内采样，航点则在半径为 10（地图尺寸的二十分之一）的圆内采样。

B. 对 *mobile-env* 的修改

在本文中，我们使用原始 mobile-env 的一个修改版本。下面介绍原始的强化学习设置：关键区别在于动作的定义。在原始设置中，动作在用户级进行定义，并不使用我们在变体中定义的每个基站阈值。这意味着，对于每个用户，动作指定连接、断开连接或不执行任何操作。每个用户的动作是 $\{0,\dots,|B_s|\}$ 中的一个整数，其中 $|B_s|$ 表示基站总数。更具体地说，0 表示不进行任何更改，而 i 会触发与基站 i 建立连接或断开连接。因此，每个用户在每个时间步最多只能触发一次连接或断开连接操作，从而使总体动作空间的大小为 $|B_s|^{|U_e|+1}$。

[2]https://github.com/Farama-Foundation/Gymnasium/blob/main/gymnasium/envs/box2d/lunar_lander.py [3]T. Haarnoja, A. Zhou, P.Abbeel, S. Levine. 软演员-评论家：具有随机演员的离策略最大熵深度强化学习。第35届国际机器学习会议论文集，PMLR 80:1861-1870，2018年。https://arxiv.org/abs/1801.01290

原始版本的状态空间包含二元用户连接指示变量，而不是我们在新版本中引入的每个基站阈值。对于每个用户，它包含：（i）一个大小为 $|B_s|$ 的二元向量，用于指示当前的连接情况；（ii）与每个基站对应的信噪比值；以及（iii）该用户先前的效用。因此，状态向量维度总计为 $|U_e| \times (2|B_s|+1)$。

两个版本中的奖励计算均保持不变

。我们决定在关联中采用逐基站阈值，而非原先的逐用户决策，主要是出于计算方面的考虑：随着用户数量增加，原始问题的动作空间会呈指数级增长，很快变得难以求解。这使得高效探索几乎不可能，也让强化学习问题变得极具挑战性。基于阈值的表述将动作空间缩减至 $3^{|B_s|}$，从而解决了这一问题；同时，由于显式连接指示变量现在完全由阈值决定，因此也简化了状态表示。此外，在实际系统中，底层关联决策的运作方式与我们在修改版本中引入的逻辑非常相似，也就是通过逐基站阈值，根据相对信号质量引导用户连接到不同的基站。

我们还修改了信噪比值的缩放方式。在原始版本中，信噪比按用户进行归一化，即每个用户的信噪比值都除以其自身的最大信噪比。我们认为，这会增加用户之间比较的难度，并可能影响策略的学习效果。由于缩放仍然很重要，我们改为引入固定的信噪比上下参考值，将数值重新缩放到这些边界范围内。上参考值采用用户位于基站附近时对应的信噪比；下参考值则采用断开连接边界，低于该边界时，mobile-env 会自动断开该用户的连接。

C. 奖励

这里，我们清楚地说明 mobile-env 中奖励计算公式的运作方式。我们用索引 i 表示基站，用索引 j表示用户。每个基站–用户对都有一个信噪比（SNR），记为 γ_{ij}。用户 j 在基站 i处的最大可实现数据速率为

$$d_{ij} = b\,\log\bigl(1+\gamma_{ij}\bigr), \tag{1}$$

其中，b 为带宽，假设所有基站和用户的带宽均相同。从调度角度来看，用户共享时隙，但不共享频率时隙（带宽）。

我们选择采用 mobile-env 中的 *RateFair* 调度策略，这意味着每个已连接用户从给定基站获得相同的数据速率。形式上，基站 i 的均等分配速率等于连接到该基站的用户数据速率的调和平均值：

$$\frac{1}{\sum_j \frac{c_{ij}}{d_{ij}}}, \tag{2}$$

其中连接指示符定义为 b y

$$c_{ij} = \mathbf{1}_{\{\gamma_{ij}\geq\tau_i\}}, \qquad c_{ij}\in\{0,1\}, \tag{3}$$

其中 τ_i 为基站 i 的连接阈值。因此，只有已连接的用户计入分母，且每个用户获得相同的份额。

用户在所有基站上的聚合速率为

$$f_j(\Gamma) = \frac{b}{n_j}\sum_i c_{ij}a_{ij}\log\bigl(1+\gamma_{ij}\bigr),$$

其中，a_{ij} 表示分配给用户 j 的来自基站 i的最大可实现数据速率 d_{ij} 的比例，$n_j=\sum_i c_{ij}$ 是连接到用户 j $\Gamma = \gamma_{ij}\ _{i\,j}\in$ 的基站数量，而 $\mathbb{R}^{|B_s|\times|U_e|}$ ( ). 是所有基站–用户对的信噪比值矩阵，其中 $|B_s|$ 和 $|U_e|$ 分别表示基站和用户的总数。

例如，在 *RateFair* 调度策略下，我们有

$$a_{ij} = \frac{1}{\sum_k c_{ik}\frac{d_{ij}}{d_{ik}}}.$$

然后将聚合速率映射为相应的效用值。

首先，应用一个效用函数：

$$g(d) = \mathrm{clip}\left(w_1\frac{\log(w_2+d)}{\log(w_3)},\, l,\, u\right),$$

参数为 w_1，w_2，w_3，l，u。接下来，使用以下方法将效用归一化到 (0，1)：

$$h(x) = \frac{x-l}{u-l}.$$

最后，在 mobile-env 中，奖励定义为所有用户的平均用户效用：

$$r = \frac{1}{|U_e|}\sum_j (h\circ g\circ f_j)(\Gamma). \tag{4}$$

D. 信道衰落

1) 其他类型的信道衰落：信道衰落被建模为用户信噪比的随机扰动，用以描述障碍物（例如建筑物）等环境因素对信号传播的影响。除了前文介绍的瑞利衰落外，我们还考虑两种莱斯衰落设置，分别为 $K=3$ 和 $K=10$。莱斯衰落模型适用于信号通过多条路径到达接收端，同时包含直达视距分量的场景。瑞利衰落可视为没有视距路径时莱斯衰落的特殊情况。此外，较大的莱斯因子 K对应较强的视距分量，因此随机性较小。

2) 衰落公式：衰落通过影响 γ（即信噪比（SNR））来影响奖励：

$$\gamma_{\text{new}} = \gamma_{\text{old}}\,|H|^2,$$

其中，H 是一个随机衰落系数，其取值来自一个分布，该分布取决于所考虑的信道衰落模型。

在这项工作中，我们研究两种经典的衰落模型：瑞利衰落和莱斯衰落，参数分别为 $K=3$ 和 $K=10$。

瑞利衰落。该模型适用于不存在视距（LoS）分量的情况，此时接收信号

由许多散射的多径分量之和产生。衰落幅度 H 随后服从瑞利分布，其概率密度函数 (pdf) 为：

$$p_H(h) = \frac{2h}{\Omega} e^{-h^2/\Omega}, \quad h \geq 0,$$

其中 $\Omega = \mathbb{E}[H^2] = 1$ 表示平均接收功率。

莱斯衰落。 当除散射多径分量外还存在直视路径时，衰落幅度 H服从莱斯分布。其概率密度函数（pdf）为：

$$f_H(h) = \frac{2(K+1)h}{\Omega} \exp\left(-K - \frac{(K+1)h^2}{\Omega}\right) I_0\left(2h\sqrt{\frac{K(K+1)}{\Omega}}\right),$$

其中 $I_0(\cdot)$ 是第一类零阶修正贝塞尔函数。参数 $K \geq 0$ 表示莱斯K因子，用于量化直视分量功率与散射分量功率之比。我们考虑两种具有代表性的情形： $K = 3$ 和 $K = 10$，其中 $\Omega = 1$。 K 值越大，表示视距条件越强；而 $K = 0$ 则对应瑞利衰落模型。

3）衰落的表述与性质： 衰落会以下述方式改变奖励。对于每个（用户，基站）对（i， j），其信道功率对应一个衰落随机变量： $|H_{ij}|^2$。新的信噪比系数为 $\gamma_{ij}\,|H_{ij}|^2$， 其中 γ 是未考虑衰落时的当前信噪比。我们将衰落下的完整信噪比矩阵记为 $\Gamma \otimes |H|^2$， 其中 $\otimes$表示逐元素乘法，并且 $\Gamma = (\gamma_{ij})_{i,\ j}$ 以及 $|H|^2 = (|H|_{ij})_{i,\ j}$。因此，回报现在是一个随机变量 R:

$$R = \frac{1}{|U_e|} \sum_j (h \circ g \circ f_j)\left(\Gamma \otimes |H|^2\right),$$

其中， $|U_e|$ 是用户数量，h、g 和 f 的定义与前一节奖励小节（第 VIII-C 节）中的定义相同。相比之下，不考虑衰落时的基准奖励是确定的：

$$r = \frac{1}{|U_e|} \sum_j (h \circ g \circ f_j)(\Gamma).$$

接下来我们证明，对于数据速率分配比例 a_{ij} 不依赖于信噪比 Γ的一大类调度策略，衰落下的期望奖励始终小于或等于基准奖励。

定理 1. 衰落下的期望奖励始终小于或等于基准奖励：

$$\mathbb{E}[R] \leq \mathbb{E}[r] = r.$$

证明。 我们先证明，对于所有用户 j，函数$h \circ g \circ f_j$是凹函数。

首先，我们证明 f_j 本身是凹函数。回想一下， f_j : $\mathbb{R}^{|B_s| \times |U_e|} \to \mathbb{R}$。函数 f_j 可以写成关于变量 γ_{ij}的可分离求和：

$$f_j(\Gamma) = \frac{b}{n_j} \sum_i \phi_{ij}(\gamma_{ij}),$$

其中 $\phi_{ij}(\gamma_{ij}) = c_{ij} a_{ij} \log\left(1 + \gamma_{ij}\right)$。

只需证明每个 ϕ_{ij} 都是凹函数，这直接源于对数函数的凹性。因此， f_j 是凹函数。最后，由于 $h \circ g$ 是凹且非递减的函数，复合函数 $h \circ g \circ f_j$ 也为凹函数。

由于对于所有用户 j，函数 $h \circ g \circ f_j$ 都是凹函数，根据詹森不等式可得

$$\begin{aligned}\mathbb{E}\left[(h \circ g \circ f_j)(\Gamma \otimes |H|^2)\right] &\leq (h \circ g \circ f_j)\left(\mathbb{E}[\Gamma \otimes |H|^2]\right) \\ &= (h \circ g \circ f_j)\left(\Gamma \otimes \mathbb{E}[|H|^2]\right) \\ &= (h \circ g \circ f_j)(\Gamma),\end{aligned}$$

其中，不等式由詹森不等式得出，第一个等号利用了 Γ 是确定性的这一事实，第二个等号则由 $\mathbb{E}[|H_{ij}|^2] = 1$ 对所有用户 j 和基站 i 均成立得出。

对所有用户取平均值可得

$$\begin{aligned}\mathbb{E}[R] &= \frac{1}{|U_e|} \sum_j \mathbb{E}\left[(h \circ g \circ f_j)(\Gamma \otimes |H|^2)\right] \\ &\leq \frac{1}{|U_e|} \sum_j (h \circ g \circ f_j)(\Gamma) = r.\end{aligned}$$

□

E. 额外衰落结果

为说明上述性质，我们报告专家数据集在不同衰落条件下的平均回报。表 V 展示了专家性能随衰落模型变化的情况。

表 V 不同衰落条件下的专家数据集。

	瑞利	莱斯 $K = 3$	莱斯 $K = 10$	无衰减
专家均值	49.01	57.79	62.64	67.74

这些数值对应于用于缩放的上界。随着随机性增加（从无衰落到瑞利衰落），专家表现会下降。这与上述证明的詹森型不等式一致：智能体在衰落条件下的期望奖励低于无衰落情况下的期望奖励。实证结果证实了这一趋势：随着衰落的随机性不断增强，表现持续下降。

表 VI 移动−环境：不同衰落模型下**高移动性**随机性设置的性能。

设置	模型	平均回报（标准差）	Δ 与无衰减相比
瑞利	DT	47.71 (65.57)	-33.8
	CGDT	80.38 (62.14)	-2.62
	CQL	**93.97 (64.35)**	**+9.17**
莱斯 $K = 3$	DT	84.18 (56.06)	+2.68
	CGDT	82.35 (55.88)	-0.65
	CQL	**89.29 (55.29)**	+4.49
莱斯 $K = 10$	DT	83.60 (53.71)	+2.10
	CGDT	77.52 (56.89)	-5.48
	CQL	**92.58 (51.31)**	+7.78

现在，我们分析莱斯衰落下的额外结果（$K = 3$ 和 $K = 10$），如表 VI 所示。CQL 仍是表现最稳定的方法，在所有设置下都取得了最高的平均回报和最低的回报方差。序列方法也表现尚可。在莱斯衰落下，决策 Transformer 受影响较小，表现也相当不错，这表明当扰动适中时，回报条件化仍能有效指导动作选择。然而，有趣的是，在这些较温和的条件下，CGDT 的表现不如决策 Transformer。这一反直觉的结果可能源于 CGDT 较高的调参复杂度——训练期间需要仔细调整非对称参数、评论家权重以及“超越数据集回报”参数——因此，相比决策 Transformer 或 CQL，它对超参数配置更为敏感。

需要指出的是，针对这种衰落类型，CGDT 方法的超参数搜索并不全面，尽管我们对四个超参数（L、τ_c、τ_p、α）测试了多种组合。这一观察也凸显出，随着随机性增加，超参数调优会变得更具挑战性。对于 CGDT 这类涉及多个调优参数的方法而言，这尤其困难；而决策 Transformer 和 CQL 在这方面则更加稳健。

IX. H移动环境的超参数选择

mobile-env 场景的超参数汇总见表 VII。

表 VII 所有 mobile−env 场景的超参数汇总。

场景	**超参数**
有限移动性	决策 Transformer：$L = 5$ CGDT：$L = 5,\quad \tau_c = 0.5,\quad \tau_p = 0.8,\quad \alpha = 0.1$ CQL：$\alpha = 1.0,\quad l_r = 3 \times 10^{-4}$ QDT：$L = 5$
高移动性	决策 Transformer：$L = 10$ CGDT：$L = 10,\quad \tau_c = 0.5,\quad \tau_p = 0.8,\quad \alpha = 0.2$ CQL：$\alpha = 0.1,\quad l_r = 3 \times 10^{-4}$ QDT：$L = 10$
瑞利 / 莱斯衰落	与高移动性相同的移动性随机性

X. 额外的移动−环境实验

A. 按回报条件绘制的图

我们考察了序列方法在有限和高移动性随机性设置下的目标回报图。这些图展示了在不同输入目标回报条件下获得的平均回报。其目的是评估序列模型遵循指定目标的能力，并最终通过拼接轨迹来达到期望的性能。

首先，我们来看图4中的有限移动性情形。这两种方法呈现出相似的趋势：起初保持平稳，随着目标回报上升而增加，随后进入第二个平稳阶段。这种表现符合直觉，因为当数据集中有足够多的轨迹达到指定的目标回报时，就可以进行拼接。然而，我们观察到，CGDT 比决策 Transformer 更紧密地跟踪目标回报。此外，在相同的目标回报下，CGDT 的表现始终优于决策 Transformer，这与其设计目标一致：实现与观测到的目标回报相当或更高的回报。

现在我们考察图5所示的高移动性随机性情况。在此设置下，两种方法都更难遵循输入目标，这证实了随机性会扰乱序列算法。不过，借助评论家，CGDT 能够在 60–65 回报区间附近更紧密地遵循目标，而该区间内可用的数据最多。这再次表明，数据稀缺时，随机性会产生更强的负面影响。

XI. 使用替代方法进行实验：QDT

表八**月球-着陆器**：确定性和随机环境下 QDT 平均回报及（标准差）与其他离线强化学习算法的比较。

模型	无随机性	高随机性
DT	90.4 (16.2)	83.5 (25.5)
CGDT	91.7 (14.5)	91.6 (19.6)
CQL	**101.2 (14.2)**	**96.9 (17.9)**
QDT	80.0 (26.2)	80.0 (23.5)

表 IX**移动-环境**：在有限与高移动性随机性条件下，QDT 平均回报及（标准差）与其他离线强化学习算法的比较。

模型	有限随机性	高随机性
DT	95.1 (9.71)	81.5 (51.6)
CGDT	**95.6 (9.75)**	83.0 (51.2)
CQL	94.6 (13.1)	**84.8 (46.9)**
QDT	95.8 (9.88)	78.3 (49.9)

我们还探索了一种结合基于贝尔曼的方法与序列式离线强化学习方法的算法，称为 Q-learning 决策 Transformer（QDT）[4]。然而，我们的实验发现，要使其达到令人满意的性能颇具挑战。为完整起见，我们介绍该算法，报告所得结果，并探讨其表现不佳的可能原因。

Q 学习决策 Transformer 通过修改轨迹回报，将 CQL 与决策 Transformer 相结合。首先，在数据集上拟合一个 CQL 模型。然后，使用保守型 Q 学习价值函数执行重新标记步骤。从每条轨迹的末尾开始，递归地重新标记回报，并使用$R_T = 0$ 和

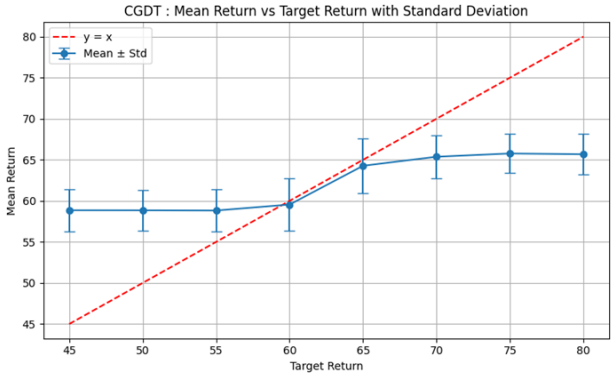
$$R_{\tau-1} \leftarrow r_{\tau-1} + \max\big(R_\tau, \hat{V}(s_\tau)\big), \tag{5}$$

其中，$\hat{V}(\cdot)$ 表示保守型 Q 学习价值函数的近似值。然而，这种重新标记可能会破坏与标准回报定义 $R_t = r_t + R_{t+1}$ 的一致性。为解决这一问题，我们引入第二种重新标记：将第一步计算出的剩余回报

[4]https://proceedings.mlr.press/v202/yamagata23a.html

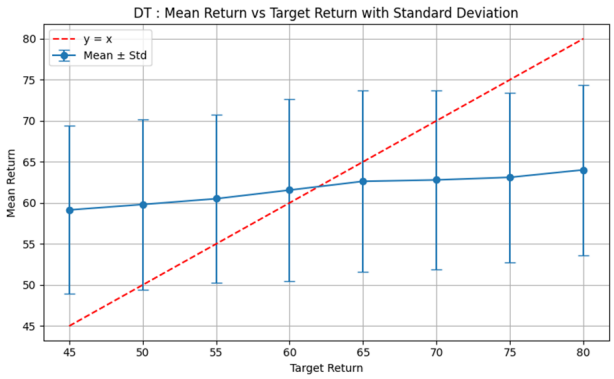


(a) 决策 Transformer

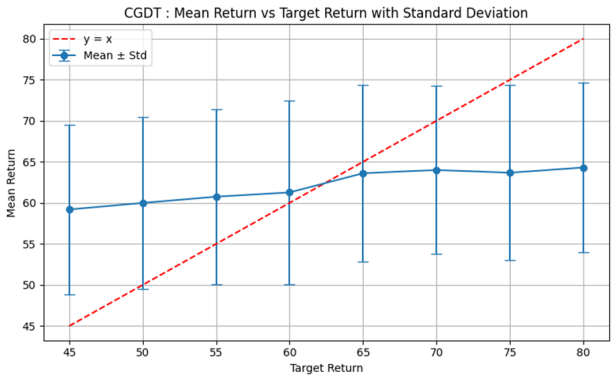


(b) CGDT

图 4 。有限移动性随机性设置下，mobile-env 中使用中等/专家数据的目标回报曲线 et.



(a) 决策 Transformer

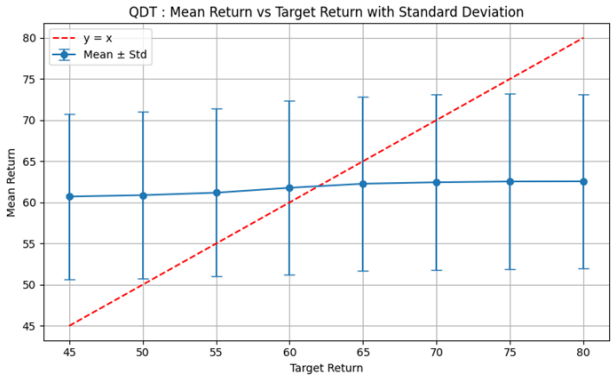


(b) CGDT

Fig. 5. 在高移动性随机性设置下，使用中等/专家数据集的 mobile-env 目标回报图 .



(a) 有限移动随机性。



(b) 高移动性随机性。

图 6. 目标回 报：使用中等/专家数据集时，QDT 在有限移动性随机性和高移动性随机性设置下的 mobile-env 图。

用作输入轨迹的最终回报，同时使用原始数据集中的奖励重新计算其余回报。本质上，与标准 DT 流程相比，这种方法只修改数据集，其余训练和推理过程保持不变。它假设价值函数通过先运行贝尔曼-

更新决策 Transformer 之前的 CQL 算法。

表 VIII 和 IX 分别总结了 QDT 在 LunarLander 和 mobile-env 环境中的结果。在 LunarLander 上，无论是在无随机性还是高随机性条件下，QDT 的表现都逊于所有其他方法。在 mobile-env 上，它在低随机性

设置下优于其他方法，但在高随机性场景中则表现落后。图 6 展示了 QDT 的目标回报曲线，凸显了它在这两种随机性条件下都难以跟上目标。

在QDT实验中，我们观察到CQL价值估计往往不准确。由于QDT依赖这些估计进行重新标记，这构成了重大挑战。虽然CQL能够正确捕捉动作的相对质量，但其绝对价值估计往往不切实际。试图通过增加CQL训练迭代次数来改进这些估计，却导致了不稳定和更差的性能。鉴于这些挑战，以及本文聚焦于CQL、决策 Transformer和CGDT，我们选择限制对QDT的研究。