用于稳定深度强化学习控制的模块化框架

Nathan P.Lawrence * **Philip D. Loewen** * **ShuyuanWang** **
Michael G.Forbes *** **R. Bhushan Gopaluni** **

*英属哥伦比亚大学数学系，加拿大不列颠哥伦比亚省温哥华，*V6T 1Z2*（电子邮箱：*input@nplawrence.com*，*loew@math.ubc.ca*）。

**英属哥伦比亚大学化学与生物工程系，加拿大不列颠哥伦比亚省温哥华，*V6T 1Z3*（电子邮箱：*antergravity@gmail.com*，*bhushan.gopaluni@ubc.ca*）

***霍尼韦尔过程解决方案，加拿大不列颠哥伦比亚省北温哥华，*V7J 3S4*（电子邮箱：*michael.forbes@honeywell.com*）

arXiv:2304.03422v1 [eess.SY] 7 Apr 2023

摘要： 我们提出了一种反馈控制器设计框架，将深度强化学习由优化驱动和无模型带来的优势，与采用 Youla-Kuˇcera 参数化定义搜索域所提供的稳定性保证相结合。行为系统领域的最新进展使我们能够构建数据驱动内部模型，从而实现一种完全基于输入输出探索数据的 Youla-Kuˇcera 参数化替代实现方式。利用神经网络表示一组参数化的非线性稳定算子，可与标准深度学习库无缝集成。我们通过对双储罐系统进行逼真的仿真，展示了该方法。

关键词：强化学习、数据驱动控制、Youla-Kučera参数化、神经网络、稳定性、过程控制

1. 引言

闭环稳定性是控制器设计中的基本要求。然而，许多基于学习的控制方案并未明确处理这一问题（Bu¸soniu 等，2018年）。这在一定程度上是可以理解的。首先，此类算法所假设的“无模型”设置，加上方法及其底层数据结构的复杂性，使得稳定性难以进行推理。其次，尤其是在强化学习（RL）的情况下，近期许多引人注目的成功案例都涉及模拟任务或游戏环境，在这些环境中，灾难性失败不会对现实世界产生实际影响。当反馈控制器需要直接通过强化学习（RL）学习时，调整折扣因子和/或奖励函数不仅会影响学习性能，还会影响探索过程中的稳定性（Bu¸soniu 等，2018年）。这一问题与现代强化学习算法的通用性和表达能力形成了对照，尽管如此，这些算法仍因控制任务而吸引了极大的关注（Nian 等，2020年）。

在这项工作中，我们提出了一种用于基于强化学习的控制器设计的保稳定性框架。我们的灵感来自 Youla-Kuˇcera 参数化（Anderson，1998），它刻画了给定系统的所有稳定化控制器。因此，关键的设计变量是一个稳定的“参数”，而不是直接的控制器表示。对稳定算子进行优化仍非易事，但我们展示了如何借助神经网络灵活地实现这一点。最后，Youla-Kuˇcera 参数化需要系统的内部模型，这与强化学习的一项关键优势相矛盾。我们利用行为系统领域的工具（Markovsky 和 D¨orfler，2021）来解决这一问题，具体而言，使用 Willems 的基本引理（Willems等，2005）。这一强有力的结果仅凭输入输出数据就能刻画系统动力学。我们利用这一点，为被控对象构建“无模型”的内部表示，从而得到与 Youla-Kuˇcera 参数化在数学上等价的实现。

总而言之，我们将基于强化学习的控制系统设计中的三个关键组成部分区分开来：算法、函数逼近器和动态模型。由此得到的框架支持采用模块化方法学习稳定策略，其中任何一个类别的进展都可用于提升整体效果。

1.1 相关工作

Bu¸soniu 等人（2018年）从控制理论视角综述了强化学习技术，并强调了对稳定性感知强化学习算法的需求。将稳定性纳入强化学习的现有策略大致可分为三类：积分二次约束（IQCs）、Lyapunov第二方法和 Youla-Kuˇcera参数化。

积分二次约束是鲁棒控制理论中的一种方法，可用于证明具有某些非线性或时变组件的动力系统的稳定性。

* ©2023 作者。本文已被 IFAC世界大会 接收，将根据知识共享许可协议 CC-BY-NC-ND 发表。

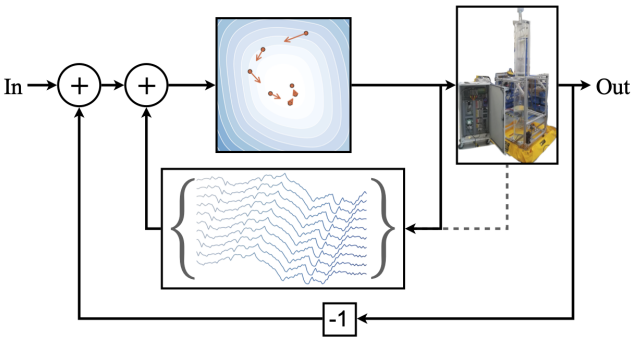

图1. 稳定的非线性参数 Q 与其环境交互；采集的输入-输出轨迹用于构造 Hankel矩阵。这些要素共同得到Youla-Kuˇcera参数化的等价实现。

在强化学习的背景下，可以使用积分二次约束来刻画环境中的非线性，或用于构成策略神经网络的非线性激活函数。近期的一些研究采用了这种方法，例如 Jin 和 Lavaei（2020年）；Wang 等人（2022年）。

Lyapunov稳定性理论在强化学习文献中也得到了充分体现（Berkenkamp等人，2017；Han等人，2020）。其核心思想是学习一个策略，以保证适当的Lyapunov函数持续下降。Berkenkamp等人（2017）提出了首批利用深度神经网络策略建立稳定性的方法之一：假设已知一个Lyapunov函数和环境的统计模型，然后在不断扩大的吸引域估计范围内优化策略。后续研究将获取Lyapunov函数这一任务纳入学习过程（Lawrence等人，2020）。例如，Han等人（2020）将可训练的Lyapunov神经网络与策略结合使用。基于融合模型预测控制与强化学习的方法（Zanon和Gros，2020）也充分运用了Lyapunov分析。

Youla-Kuˇcera参数化是一种看似未被充分利用的将稳定性融入强化学习算法的技术。Friedrich和Buss（2017）通过使用粗略的被控对象模型来采用Youla-Kuˇcera参数化；强化学习被用于以安全的方式优化物理双自由度机器人的跟踪性能，同时考虑未建模的非线性。最近，研究人员开发了一种基于积分二次约束的循环神经网络架构（Revay等，2021年）。由于该架构在设计上满足稳定性条件，因此可以用于Youla-Kuˇcera参数化的非线性版本中的控制（Wang等，2022年）。

虽然我们也采用 Youla-Kuˇcera参数化，但我们的方法有几个创新之处。我们使用非循环神经网络结构来生成稳定算子；这使得该方法易于实现，也便于与现成的强化学习算法集成，适用于在策略和离策略学习。我们还基于Willems基本引理，提出了Youla-Kuˇcera参数化的数据驱动实现，从根本上消除了先验建模假设。

2. 背景

本节介绍我们的方法所需的基础内容。我们首先将Willems引理与Youla-Kuˇcera参数化联系起来，然后展示我们学习稳定算子的方法。

我们考虑以下形式的线性时不变（LTI）系统

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t + Du_t \end{aligned} \tag{1}$$

有时，将方程 (1) 表示为传递函数会很方便，此时我们写作 $P = P($ $z = C\ zI - A\ ^{-1}B + D$ $A\ B$ $)\ ()$。我们假设 (,) 是可控的，且 (A, C) 是可观的，并且已知系统阶数的一个上界。至关重要的是，系统矩阵未知。为简化表述，我们假设所研究的系统是稳定的单输入单输出系统，不过，这些结果可以扩展到更一般的情况。

2.1 Youla-Kuˇcera参数化的数据驱动实现

给定一个由 N个 $\mathbb{R}^m$中的向量构成的序列 $\{z_t\}_{t=0}^{N-1}$，阶数为 L 的*Hankel*矩阵定义为

$$H_L(z) = \begin{bmatrix} z_0 & z_1 & \cdots & z_{N-L} \\ z_1 & z_2 & \cdots & z_{N-L+1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{L-1} & z_L & \cdots & z_{N-1} \end{bmatrix}.$$

定义 *1.* 序列 $\{z_t\}_{t=0}^{N-1} \subset \mathbb{R}^m$ 是阶数为 L 的持续激励序列，当且仅当 rank$(H_L(z))\ = mL$。

定义 *2.* 若存在状态序列 $\{x_t\}_{t=0}^{N-1}$使得方程（1）成立，则输入输出序列 $\{u_t, y_t\}_{t=0}^{N-1}$是一个 LTI 系统（A, B, C, D）的轨迹。

以下定理是 Willems基本引理（Willems等，2005）的状态空间版本。它完全基于输入输出数据，为 LTI 系统提供了一种替代性刻画。只需要系统阶数的上界。

定理 *3.* （参见 van Waarde等（2020））。设 $\{u_t, y_t\}_{t=0}^{N-1}$是一个 LTI系统（A，B，C，D）的轨迹，其中 $\underline{u}$ 是 $L + n$ 阶持续激励的。那么，$\{u_t, y_t\}_{t=0}^{L-1}$是（A，B，C，D）的轨迹，当且仅当存在 $\alpha \in \mathbb{R}^{N-L+1}$ 使得

$$\begin{bmatrix} H_L(u) \\ H_L(y) \end{bmatrix} \alpha = \begin{bmatrix} \overline{u} \\ \overline{y} \end{bmatrix}. \tag{2}$$

（当我们在等式 (2) 右侧的上下文中省略时间索引时，它被理解为列向量 $\underline{z} = [z_0 \ldots z_{L-1}]^T$。）定理 3 已广泛应用于预测控制任务（Markovsky 和 D¨orfler，2021年；Berberich和Allgower，2020年）。具体而言，可以提出一系列输入，并通过对等式 (2) 做少许改动，计算出相应的输出。这形成了一种预测输入序列并“补全”输出的方案。

接下来，我们考虑将基于 Hankel 的模型用作内部系统模型的情形。因此，我们

假设系统是严格真有理的——也就是说， $D=0$ 在方程 (1) 中——以确保之后的控制器策略可实现。现在，给定一条系统轨迹 $\{u_t,y_t\}_{t=0}^{L-1}$，我们注意到 y_L 由这些可用数据唯一确定。向前推进系统的一种简单方法是考虑一个时间移位的Hankel矩阵

$$H'_L(z)=H_L(z'),$$

其中 $z=\{z_t\}_{t=0}^{N-1}$ 和 $z'=\{z_t\}_{t=1}^{N}$，对于某个 N。

推论*4*。令 $\{u_t,y_t\}_{t=0}^{N-1}$为一个严格真有理 LTI系统（A， B， C）的轨迹，其中 u 是 $L+n+1$阶持续激励的。则对于（A， B， C）的每条轨迹 $\{u_t,y_t\}_{t=0}^{L-1}$，存在 $\alpha\in\mathbb{R}^{N-L}$使得

$$\overline{y}'=H'_L(y)\alpha.$$

证明。 根据定理 3，轨迹 $\{u_t,y_t\}_{t=0}^{L-1}$ 满足

$$\begin{bmatrix}H_L(u)\\H_L(y)\end{bmatrix}\alpha=\begin{bmatrix}\overline{u}\\\overline{y}\end{bmatrix}$$

对于某些 $\alpha\in\mathbb{R}^{N-L}$。此外，根据定义 2，存在一个状态序列 $\{x_t\}_{t=0}^{L-1}$，该序列对应于输入-输出轨迹 $\{u_t,y_t\}_{t=0}^{L-1}$。此序列诱导出状态 $\overline{x}_L$。我们有

$$\begin{aligned}\overline{y}_L&=C\overline{x}_L\\&=C\left(A\overline{x}_{L-1}+B\overline{u}_{L-1}\right)\\&=C\left(A\sum_{i=0}^{N-L-1}\alpha_i x_{L+i}+B\sum_{i=0}^{N-L-1}\alpha_i u_{L+i}\right)\\&=\sum_{i=0}^{N-L-1}\alpha_i C\left(Ax_{L+i}+Bu_{L+i}\right)\\&=\sum_{i=0}^{N-L-1}\alpha_i y_{L+i+1}\end{aligned}$$

如预期。□

推论 4给出了一种系统地将轨迹随时间向前推进的方法。在在线实施反馈控制器的同时，这对于将真实系统与汉克尔表示对齐尤其有用。

Youla-Kuˇcera参数化通过内部系统模型与稳定算子的组合，生成所有稳定化控制器的集合。其关键在于直接对与被控对象相关的闭环传递函数进行参数化，然后恢复出控制器。例如，闭环传递函数 $\frac{PC}{1+PC}$从参考信号 r到输出 y 的行为由传递函数 $\frac{C}{1+PC}$决定。通过引入稳定的设计变量 Q，我们便可以通过传递函数 PQ直接塑造系统的稳定行为。令 $Q=\frac{C}{1+PC}$，即可得到Youla-Kuˇcera参数化（Anderson，1998）：

$$\mathcal{C}_{\text{stable}}=\left\{\frac{Q}{1-QP}\colon Q\text{ is stable}\right\}$$

这一结果还可进一步推广到不稳定的多输入多输出系统。此外，当 P 为线性时，可以使用非线性算子 Q (Anderson, 1998)。

算法 1 数据驱动的稳定控制 er

输入： 稳定 Q 参数；观测值 $\{u_k,y_k\}_{k=0}^{N-1}$；初始轨迹 $\{u_k,y_k\}_{k=0}^{L-1}$

对于每个时间步长 t **执行**

设置 $u_{t-1}\leftarrow\overline{u}_{L-1}$

观察跟踪误差 $e_t=r_t-y_t$

根据推论 4 计算 y_L

将输入 $\widehat{r}=e_t+y_L$ 应用于 Q 参数并返回控制动作 u_L

更新轨迹：

$$\{\overline{u}_k,\overline{y}_k\}_{k=0}^{L-1}\leftarrow\{\overline{u}_k,\overline{y}_k\}_{k=1}^{L}$$

在算法 1 中，我们将上述数学思想转化为一个直接的顺序过程。以下结果详细说明了两者之间的对应关系。

定理 *5*。 假设 P 是一个稳定且严格真有理的LTI系统。令 Q为一个稳定且真有理的 LTI 参数。给定 P阶数的上界 L ，算法 1生成与 Youla- Kuˇcera 参数化相同的控制信号 $\{u_t\}_{t=0}^{\infty}$ 。

证明。我们使用 q_t, p_t 分别表示 Q 和 P的脉冲响应。类似地，相应的最小状态空间矩阵表示为 (A_q,B_q,C_q,D_q) 和(A_p,B_p,C_p)。

根据 Youla-Kuˇcera参数化，我们有

$$\begin{aligned}&C(z)=\frac{Q(z)}{1-Q(z)P(z)}\quad\forall z\in\mathbb{C}\\\iff\quad&\left(1-Q(z)P(z)\right)U(z)=Q(z)E(z)\\\iff\quad&u_t=q_t*(e_t+p_t*u_t)\quad\forall t\in\mathbb{N}_0\\&\qquad=\sum_{j=0}^{t-1}C_qA_q^{t-1-j}B_q\widehat{r}_j+D_q\widehat{r}_t,\end{aligned}\tag{3}$$

其中 $\widehat{r}_j=e_j+\sum_{i=0}^{j-1}C_pA_p^{j-1-i}B_pu_i$ 和 $*$ 是卷积算子；我们还在不失一般性的情况下假设， P 和 Q 的初始状态为零。

接下来，我们将公式 (3) 与算法 1 联系起来。令 $\{e_k\}_{k=0}^{\infty}$ 为任意序列。（在算法 1 中，这样的序列是动态生成的。）不失一般性，令初始轨迹为 $\{u_k,y_k\}_{k=0}^{L-1}=\{0,0\}_{k=0}^{L-1}$。对于每个时刻 $t\in\mathbb{N}_0$，我们根据推论 4 计算 $\alpha^{(t)}$ 和 $y_t=y_L$。由于 L 是 P 的阶的上界，因此 y_t 是轨迹 $\{u_k,y_k\}_{k=0}^{L-1}$ 的唯一下一个输出。因此，我们有

$$\widehat{r}_t=e_t+\sum_{i=0}^{N-L-1}\alpha_i^{(t)}y_{L+i+1}.$$

Then

$$\overline{u}_t=\sum_{j=0}^{t-1}C_qA_q^{t-1-j}B_q\widehat{r}_j+D_q\widehat{r}_t$$

给出下一个控制输入。

通过在时间步长之间更新轨迹——$\{u_k,y_k\}_{k=0}^{L-1}\leftarrow\{u_k,y_k\}_{k=1}^{L}$——我们动态生成序列$\{\alpha^{(t)}\}_{t=0}^{\infty}$，该序列生成的控制输入 $\{u_t\}_{t=0}^{\infty}$ 满足方程（3）中的离散积分方程。□

2.2 学习稳定算子

Youla-Kučera参数化非常优雅，因为它将任何问题的搜索空间细化为稳定算子集合。然而，在该集合上进行有效优化仍然是一项重大挑战（Wang 等，2022）。

我们采用 Lawrence等人（2020）提出的方法。首先回顾一下*Lyapunov*候选函数 V的概念：$\mathbb{R}^n \to \mathbb{R}$：1) V连续；2) 对所有 $z \neq 0$， $V(z) > 0$，且 $V(0) = 0$；3) 存在连续且严格递增的函数 φ：$[0,\infty) \to [0,\infty)$，使得对于所有 $z \in \mathbb{R}^n$，$V(z) \geq \varphi(\|z\|)$；4) 当 $\|z\| \to \infty$时， $V(z) \to \infty$。Lyapunov函数对于通过 Lyapunov第二方法（Khalil, 2002）证明系统稳定至关重要。Lawrence等人（2020）通过使用可训练Lyapunov函数，“按设计”构造了形式为 $z_{t+1} = f_\theta(z_t)$的稳定自治系统。满足上述主要要求的神经网络可以通过稍作修改的输入凸神经网络（Amos等人，2017）构建；详情请参阅 Lawrence等人（2020）及其中的参考文献。

两个神经网络协同工作，组成一个满足$\widehat{}$Lyapunov第二方法核心下降条件的单一模型：一个光滑神经网络 f_θ，以及一个Lyapunov神经网络 V_θ。令 $\hat{z}' = f_\theta(z)$，其中 z 是当前状态， $\hat{z}'$ 是建议的下一状态。可能出现两种情况：$\hat{z}'$ 降低了 V 的值，或者没有。利用 V的凸性，我们可以用闭式形式写出对动力学的“修正”：

$$
\begin{aligned}
z_{t+1} &= f_\theta(z_t) \\
&\equiv \begin{cases} \widehat{f}_\theta(z_t), & \text{if } V(\widehat{f}_\theta(z_t)) \leq \beta V(z_t) \\ \widehat{f}_\theta(z_t)\left(\frac{\beta V(z_t)}{V(\widehat{f}_\theta(z_t))}\right), & \text{otherwise} \end{cases} \\
&= \gamma \widehat{f}_\theta(z_t), \text{ where} \\
\gamma = \gamma(z_t) &= \frac{\beta V(z_t) - \texttt{ReLU}\big(\beta V(z_t) - V(\widehat{f}_\theta(z_t))\big)}{V(\widehat{f}_\theta(z_t))}.
\end{aligned} \tag{4}
$$

$\widehat{}$由于方程（4）定义了模型 f_θ，因此 f_θ 和 V_θ 会同步训练，以实现序列状态 z_t， z_{t+1}，.....所需的任意目标，例如监督学习任务。此外，尽管模型 f_θ 受到稳定性约束，但其参数空间不受约束，因此使用深度学习库实现和训练相当简单。此外，尽管方程（4）的结构复杂，Lawrence等人（2020年）证明了整体模型是连续的。这里，我们使用 f_θ 对非线性 Q 参数的内部动力学进行建模。例如，可以使用具有稳定转移动力学的控制仿射模型 f_θ。

3. 无约束稳定强化学习

下面简要概述深度强化学习，以定义我们的符号；这些符号大多是标准用法。有关更多背景信息，请参阅 Sutton 和 Barto（2018年）以及 Bu‚soniu 等人（2018年）；Nian 等人（2020年）则提供了教程式的介绍。

强化学习是一个优化驱动的框架，仅通过与环境交互来学习“策略”（Sutton 和 Barto，2018年）。状态s 和动作 a 分别属于状态集和动作集 $\mathcal{S}$、 $\mathcal{A}$。在每个时间步 t，状态 s_t 影响从“策略” π中采样动作 $a_t \sim \pi(\cdot \mid s_t)$。给定动作 a_t后，环境生成后继状态 s_{t+1}。这一循环完成马尔可夫决策过程中的一个步骤，对于任意初始分布 $s_0 \sim p_0(\cdot)$，该过程会诱导出条件密度函数$s_{t+1} \sim p(\cdot \mid s_t,\ a_t)$。随着时间步在策略 π下推进，“轨迹”记作$h = (s_0,\ a_0,\ r_0,\ s_1,\ a_1,\ r_1,\ \ldots)$。每个固定策略 π都会在轨迹集合上诱导出概率密度 $p^\pi(\cdot)$。

在强化学习中，每个阶段的“奖励” $r_t = r(s_t,\ a_t)$用于量化一次给定轨迹的优劣。智能体的总体目标是确定一个策略，使累积折扣奖励最大化。也就是说，给定某个常数 $\gamma \in (0,\ 1)$，

$$
\begin{aligned}
&\text{maximize} \quad J(\pi) = \mathbb{E}_{h\sim p^\pi}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right] \\
&\text{over all} \quad \text{policies } \pi\colon \mathcal{S} \to \mathcal{P}(\mathcal{A}),
\end{aligned} \tag{5}
$$

其中， $\mathcal{P}(\mathcal{A})$表示 $\mathcal{A}$上的概率测度集合。

在所有可能的策略空间中，优化是在由某个向量 θ参数化的子集上进行的。例如，在某些应用中， θ 表示深度神经网络中的所有权重集合。在本研究中，策略是第 2.2 节所述的非线性 Q 参数。因此，公式 (5) 自动满足整个权重空间 θ上的内部稳定性约束。由此，我们可以使用任意强化学习算法来求解该问题。常用强化学习算法的内部原理已有详尽记载（Nian 等，2020），因此本文不再赘述，而仅作简要概述。

强化学习的广泛主题涉及用于选择理想策略 π 的迭代方法（这就是“学习”），并以某种根本方式由智能体对过去状态—动作对所获奖励的观察来指导（这提供了“强化”）。

解决问题（5）的标准方法是使用梯度上升

$$
\theta \leftarrow \theta + \alpha \nabla J(\theta),
$$

其中 $\alpha > 0$ 是步长参数。对于随机策略和确定性策略，$\nabla J(\theta)$ 均有解析表达式（参见 Sutton 和 Barto（2018年）；Bu‚soniu 等人（2018年））。关键的是，这些公式依赖于状态-动作价值函数[1],

$$
Q^{(\mathrm{RL})}(s_t, a_t) = \mathbb{E}_{h\sim p^\pi}\left[\sum_{k=t}^{\infty} \gamma^{k-t} r(s_k, a_k) \middle| s_t, a_t\right].
$$

尽管 $Q^{(\mathrm{RL})}$ 并不确切，因为它同时取决于动力学和策略，但可以使用深度神经网络对其进行估计（Bu‚soniu等，2018年）。这些思想和各种近似技术构成了深度强化学习算法的基础。

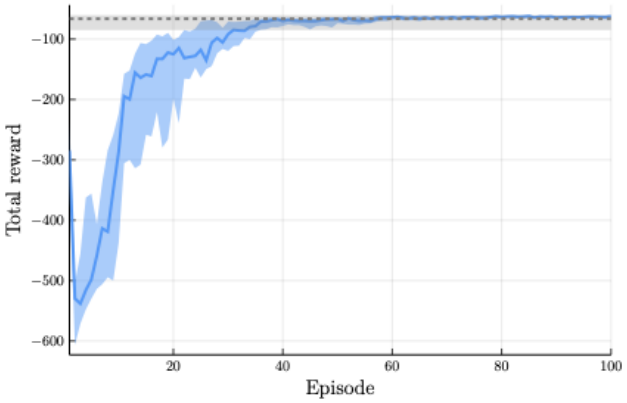

图2. 20个训练会话中的累积奖励曲线。实线表示中位数，阴影区域表示四分位距。虚线及其阴影区域表示未施加稳定性约束时的最终训练结果。

4. 仿真示例

我们在一个涉及两个大型储罐的液位控制系统的逼真仿真中展示了训练结果。目标是调节上部储罐中的水位，同时水会持续排入下部储水池。泵将水从储水池提升回上部储罐，从而形成循环流动。该装置的实物示意图见图1，Lawrence 等（2022）对此作了进一步说明。

系统动力学基于伯努利方程，确定流出流量 $f_{\text{out}} \approx f_c\sqrt{2g\ell}$，以及上部储罐中的流体体积守恒：

$$\frac{d}{dt}\left(\pi r_{\text{tank}}^2 \ell\right) = \pi r_{\text{tank}}^2 \dot{\ell} = f_{\text{in}} - f_{\text{out}}$$

（我们使用点号表示相对于时间的微分； g 是引力常数； ℓ 是液位； r_{tank} 是储罐的半径；有关该系统的更详尽描述，请参见 Lawrence等（2022）。）我们的应用涉及四个滤波信号，其时间常数分别为：泵的 τ_p 、流入流量变化的 τ_{in} 、流出流量的 τ_{out}以及测量液位动态的 τ_m 。因此，我们得到以下描述泵速、流量、液位和测量液位的微分方程系统：

$$\begin{aligned}
\tau_p \dot{p} + p &= p_{\text{sp}} \\
\tau_{\text{in}} \dot{f}_{\text{in}} + f_{\text{in}} &= f_{\max}\left(\frac{p}{100}\right) \\
\tau_{\text{out}} \dot{f}_{\text{out}} + f_{\text{out}} &= \pi r_{\text{pipe}}^2 f_c \sqrt{2g\ell} \\
\pi r_{\text{tank}}^2 \dot{\ell} &= f_{\text{in}} - f_{\text{out}} \\
\tau_m \dot{m} + m &= \ell
\end{aligned}$$

为了跟踪期望液位 ℓ_{sp} ——“sp”代表“设定值”——我们可以通过加入以下方程来使用液位和流量控制器：

$$\begin{aligned}
p_{\text{sp}} &= \text{PID}_{\text{flow}}(f_{\text{in,sp}} - f_{\text{in}}) \\
f_{\text{in,sp}} &= \text{PID}_{\text{level}}(\ell_{\text{sp}} - m)
\end{aligned} \tag{6}$$

方程 (6) 使用简写分别表示以误差信号 $f_{\text{in, sp}} - f_{\text{in}}$ 和 $\ell_{\text{sp}} - m$ 为输入的 PID 控制器。对于

[1] 遗憾的是，这个函数和 Youla-Kučera 参数通常都用 Q 表示。这也解释了这里的上标。

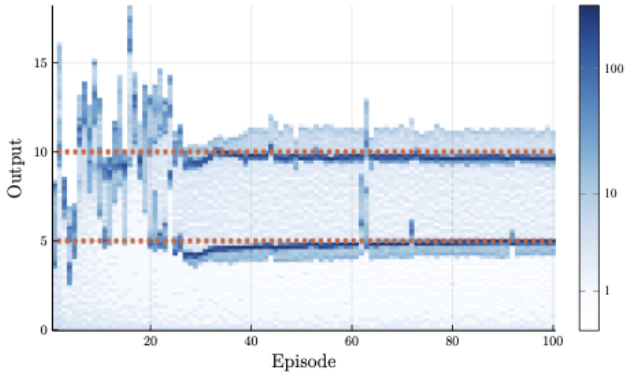

图 3. 全部 20 次会话训练进度的全局视图。对于每个回合，统计在不同输出值上花费的时间分布。热图显示了在每个回合–输出坐标上花费的平均时间。

在我们的研究中，PID_{flow} 和 $\text{PID}_{\text{level}}$ 是固定的，且属于环境的一部分。这些动力学以 0.5 秒的离散时间步长实现，并带有方差为 0.015 的高斯测量噪声。

（训练结果）由于环境包含 PID 控制器，我们将控制方案修改为增量形式 $u_t = u_{t-1} + \Delta u_t$，其中 Δu_t 是 Youla-Kučera 参数与 PID 控制器输出之和：

$$\Delta u_t = \Delta u_t^{(q)} + \Delta u_t^{(\text{PID})}$$

我们进行了 20 个训练会话，每个会话包含 100 个回合，并将结果汇总于图 2 和图 3。图 2 展示了每个回合的累计奖励。我们呈现了 20 个训练会话的中位数和四分位距；可以看到，中位数奖励曲线明显更接近阴影区域的上限，而非下限，这表明大多数实验结果都落在该较窄的区域内。尽管由于随机策略初始化，初始阶段存在显著差异，但各训练会话均表现出一致的收敛趋势。奖励曲线通常在约 40 个回合后进入平台期。

图 3 展示了各回合在整个训练会话中的集体演变情况。每个回合–输出坐标的阴影深浅取决于输出变量平均在该位置停留的时间。虚线值（设定值）附近的颜色越深越理想。该图旨在粗略说明图 2 中奖励曲线所代表的含义。另一方面，图 4 展示了其中一项实验的一次单次运行。

5. 结论

Youla-Kučera参数化在控制理论中广为人知，但在强化学习中似乎未得到充分利用。以此为出发点，我们采用了深度学习和行为系统方面的进展，开发出一个端到端框架，用于通过通用强化学习算法学习稳定化策略。本文是概念验证，仍有许多值得探索的方向，包括使用随机策略、扩展到不稳定系统，以及平衡训练期间与稳态运行期间的持续激励假设。我们认为这是一个富有成果的研究领域。

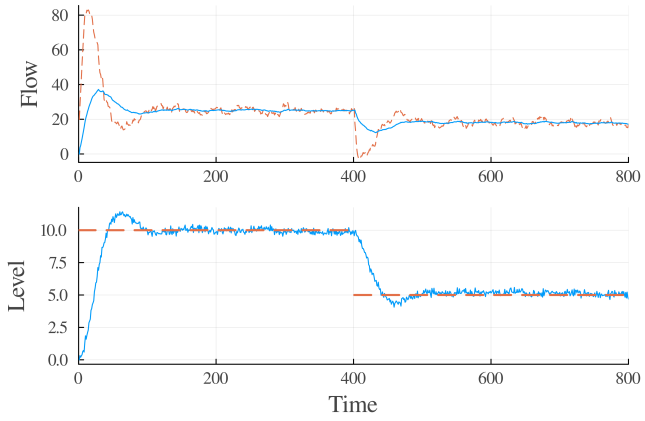

图 4. 训练后的强化学习智能体在其中一个训练回合中的输入输出示例。虚线为设定值；实线为测量值。

以便随着深度强化学习在过程系统工程中逐渐受到重视而开展进一步研究。

致谢

我们衷心感谢加拿大自然科学与工程研究委员会（NSERC）和 Honeywell Connected Plant 提供的资金支持。我们也要感谢 Yaniv Plan 教授的宝贵讨论。

参考文献

Amos, B., Xu, L. 和 Kolter, J.Z.（2017）。输入凸神经网络。载于 *Proceedings of the 34th International Conference on Machine Learning* 70 *Proceedings of Machine Learning*，第 卷，–。*Research* 146 155 https://proceedings.mlr.press/ PMLR。网址：v70/amos17b.html 。

Anderson, B.D.（1998）。从 Youla–Kucera 到辨识、自适应与非线性控制。*Automatica*，34(12)，1485–1506。doi：10.1016/S0005-1098(98)80002-2。

Berberich, J. 和 Allgower, F.（2020）。用于数据驱动系统分析与控制的基于轨迹的框架。载于 *Control Conference (ECC)* 1365 1370 *2020 European*，–。电气电子工程师学会，俄罗斯圣彼得堡。doi：10.23919/ECC51009.2020.9143608。

Berkenkamp, F.、Turchetta, M.、Schoellig, A. 和 Krause, A.（2017）。具有稳定性保证的安全基于模型强化学习。载于 *Advances in Neural Information Processing Systems*，第 30 https://proceedings.neurips.cc/paper/2017/file/ 卷。网址：。766ebcd59621e305170616ba3d3dac32-Paper.pdf

Bu.soniu, L.、de Bruin, T.、Toli´c, D.、Kober, J. 和 Palunko, I.（2018）。用于控制的强化学习：性能、稳定性与深度逼近器。 *Annual Reviews in Control*，46，8–28。doi：10.1016/j.arcontrol.2018.09.005。Friedrich, S.R. 和 Buss, M.（2017）。一种基于稳定化控制器参数化的机器人强化学习鲁棒稳定性方法。载于 *2017 IEEE International Conference on Robotics and Automation (ICRA)* 3365 3372 *IEEE*，–。电气电子工程师学会，新加坡，新加坡。doi：10.1109/ICRA.2017.7989382。Fujimoto, S.、van Hoof, H. 和 Meger, D.（2018）。解决演员-评论家方法中的函数逼近误差。载于 *Proceedings of the35th International Conference on Machine Learning*，*Proceedings of Machine Learning Research* 第 80 卷，1587–1596。PMLR。网址：https://proceedings.mlr.press/v80/fujimoto18a.html。Han, M.、Zhang, L.、Wang, J. 和 Pan, W.（2020）。具有稳定性保证的用于控制的演员-评论家强化学习。*Robotics and Automation Letters* 5 4 6217 6224 10 1109 ，（），–。doi：。/ LRA.2020.3011351。

Jin, M. 和 Lavaei, J.（2020）。稳定性认证的强化学习：控制理论视角。*IEEE Access : practical innovations open solutions* 8 229086 229100 10 1109 ，，，–。doi：。/ ACCESS.2020.3045114。Khalil, H.K.（2002）。 *Nonlinear Systems*。Prentice-Hall。Lawrence, N.P.、Forbes, M.G.、Loewen, P.D.、McClement, D.G.、Backstr¨om, J.U. 和 Gopaluni, R.B.（2022）。采用浅层控制器的深度强化学习：PID 调节的实验性应用。 *Control Engineering Practice*，121，105046。doi：10.1016/j.conengprac.2021.105046。Lawrence, N.P.、Loewen, P.D.、Forbes, M.G.、Backstr¨om, J.U. 和 Gopaluni, R.B.（2020）。几乎必然稳定的深度动态系统。载于 *Advances in Neural Information Processing Systems* 33， 18942 18953 ，第 – 卷。Curran Associates, Inc. 网址：https://proceedings.neurips.cc/paper/2020/file/daecf755df5b1d637033bb29b319c39a-Paper.pdf 。

Markovsky, I. 和 D¨orfler, F.（2021）。数据驱动分析、信号处理与控制中的行为系统理论。 *Annual Reviews in Control* 1367578821000754 10 1016 ，S。doi：。/ j.arcontrol. 2021.09.005。Nian, R.、Liu, J. 和 Huang, B.（2020）。强化学习综述：工业过程控制中的简介与应用。*Computers & Chemical Engineering*，139，106886。doi：10.1016/j. compchemeng.2020.106886。Revay, M.、Wang, R. 和 Manchester, I.R.（2021）。循环平衡网络：具有稳定性与鲁棒性保证的灵活动态模型。doi：10.48550/ARXIV.2104.05942。Sutton, R.S. 和 Barto, A.G.（ 2018 *Reinforcement Learning: An* ）。。《自 *Introduction* 适应计算与机器学习丛书》。麻省理工学院出版社，马萨诸塞州剑桥，第二版。Tian, J. 和贡献者 o.（2020）。ReinforcementLearning.jl：用于 Julia 编程语言的强化学习软件包。网址：https://github.com/JuliaReinforcementLearning/ReinforcementLearning.jl 。van Waarde, H.J.、De Persis, C.、Camlibel, M.K. 和 Tesi, P.（2020）。适用于状态空间系统的 Willems基本引理及其向多数据集的扩展。 *IEEE Control Systems Letters*，4 3 602 607 10 1109 2020 2986991（），–。doi：。/LCSYS. . . Wang, R.、Barbara, N.H.、Revay, M. 和 Manchester, I.R.（2022）。针对部分可观测线性系统，在所有稳定非线性控制器上进行学习。 *IEEE Control Systems Letters*，7，91–96。doi：10.1109/LCSYS.2022.3184847。Willems, J.C.、Rapisarda, P.、Markovsky, I. 和 De Moor, B.L.（2005）。关于激励持久性的一则说明。 *Systems & Control Letters*，54 4 325 329 10 1016 2004 09 003（），–。doi：。/j.sysconle. . . . Zanon, M. 和 Gros, S.（2020）。使用鲁棒 MPC 的安全强化学习。 *IEEE Transactions on Automatic Control*，66(8)，3638–3652。doi：10.1109/TAC.2020.3024161。

附录 A. 实现细节

数值实验使用 Julia 编程语言进行。具体而言，我们使用了 ReinforcementLearning.jl软件包（Tian 和贡献者，2020年）。我们使用 TD3算法（Fujimoto 等，2018年）更新网络参数。我们使用了奖励函数 $-0.1|y_{\mathrm{sp}}-y|-0.01\left(\Delta u^{(q)}\right)^2$。大多数超参数均采用默认值；我们将策略延迟设为 4。我们始终使用双层前馈网络。评论家网络每层有64 个节点，并使用 softplus激活函数。在图2 所示的无稳定性对比中，我们对演员使用了相同的结构。对于稳定的 Q 参数，我们使用公式 (4)，并使用矩阵为 B, C, D 和 $\widehat{f_\theta}$的状态空间模型，而不是标称的 A 矩阵。 f_θ 每层有 16 个节点，并使用 tanh 激活函数。 V_θ 每层也有 16 个节点。