

# 储备池算法学习过程与稳定性分析

蓝秀文<sup>1</sup>, 陈伟<sup>2</sup>, 高健<sup>1</sup>, 颜子翔<sup>1</sup>, 肖井华<sup>1</sup>

(1.北京邮电大学 理学院; 数学与信息网络教育部重点实验室, 北京 100876;

2.安徽科技学院 信息与网络工程学院, 安徽 滁州 233100)

**摘要:** 储备池计算已广泛应用于学习和预测动态行为, 可以实现包括混沌系统在内的复杂时序信号的演化预测. 然而, 储备池如同大部分神经网络算法一样, 其学习过程被认为是黑盒子, 缺乏对其系统性的研究和解释. 本文通过建立最小单节点储备池计算模型, 将学习与预测过程作为一种非线性动力系统进行分析. 研究发现, 训练序列与输出序列之间存在复杂的映射关系, 且与待学习系统本身的动力学分岔性质密切相关. 通过稳定性分析, 得到了输入和输出状态之间对应关系的相图, 揭示了预测失败现象对应的储备池系统稳定性的变化. 在单节点系统分析的基础上, 进一步发现了随着储备池规模的增大, 预测失败概率的快速降低, 以及其对应的储备池稳定参数区域的迅速增大. 本文以 logistic 映射对应的周期与混沌信号为例, 理论分析与数值结果完全对应. 通过储备池学习过程与稳定性的分析, 为算法在学习和预测各种动态行为中的成功提供了动力学视角的理论基础.

**关键词:** 储备池算法; 机器学习; 非线性动力学; 稳定性分析

**中图分类号:** O415

**文献标志码:** A

**文章编号:** 1000-2367(2026)02-0124-11

从时间序列中学习和推断系统动态是研究复杂系统的重要方法之一. 基于时间序列的分析方法广泛应用于天气和气候预测、金融数据分析等研究领域, 并在控制与识别、适应性滤波、机器人技术、视觉与语音的生产和处理等方面起到重要作用<sup>[1]</sup>. 最近, 随着人工智能和神经网络方法的快速发展, 一类新型时间序列分析方法, 循环神经网络方法(recursive neural network, RNN)被广泛应用于复杂系统的分析中, 包括机器识别、武器系统控制, 以及语音识别等问题<sup>[2-4]</sup>.

随着循环神经网络方法技术的发展, 也产生了如长短时记忆网络等基于循环神经网络的计算方法. 文献[5-8]分别通过回声状态网络和液态状态机提出一种新的实用计算方法, 并在之后被统称为储备池计算(reservoir computing, RC). 作为循环神经网络的简化方案, 在输入层、神经网络计算层、输出层中, 储备池算法仅保留了对输出层权重的训练. 与循环神经网络相比, 储备池算法具有结构简单、学习速度快、计算成本低等优点, 并且擅长处理由动力系统生成的时间序列数据<sup>[9]</sup>. 因此, 储备池算法广泛应用于复杂动力系统的学习与预测, 如预测混沌吸引子、广义同步、密码学等<sup>[10-15]</sup>.

在广泛应用的同时, 储备池算法作为神经网络算法, 其本身也具有神经网络的黑盒子属性. 例如, 在相同的训练数据条件下, 算法参数的不同会使得预测结果发生不可预测的偏移. 这对于神经网络算法的使用带来了不可控的风险. 能否理解储备池算法的内在工作机理, 分析预测结果偏移的内在机制, 对于储备池算法的理论研究以及实际应用都具有重要意义<sup>[16-19]</sup>.

收稿日期: 2025-02-22; 修回日期: 2025-03-27.

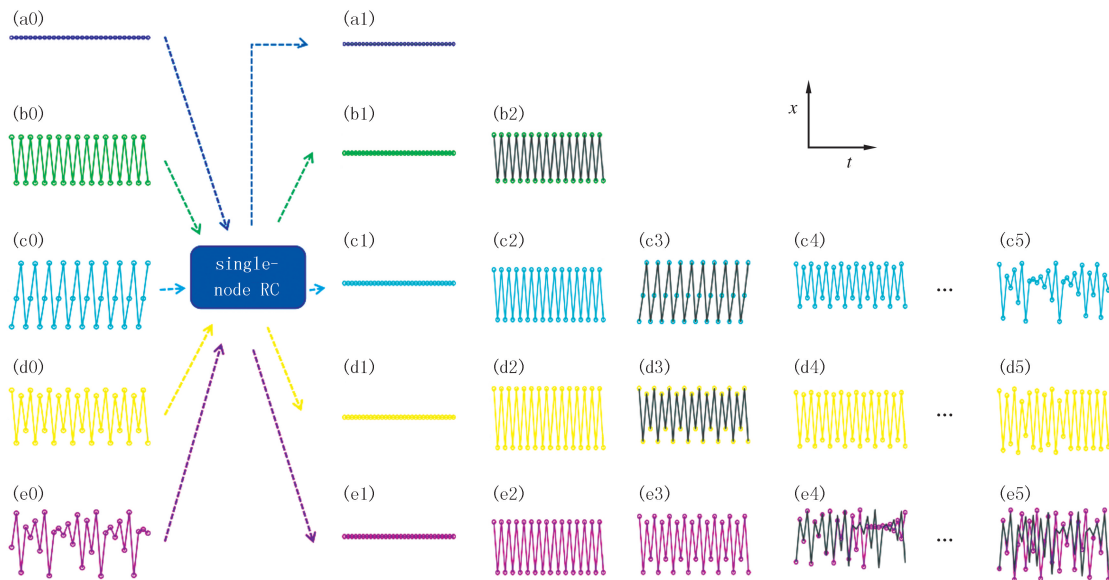
基金项目: 国家自然科学基金(11775034); 安徽省高校科研项目(2024AH050334).

作者简介: 蓝秀文(1996-), 女, 湖北黄冈人, 北京邮电大学硕士研究生, 研究方向为储备池网络学习机制, E-mail: lanxiuwen1204@163.com.

通信作者: 高健, E-mail: gao.jian@bupt.edu.cn.

引用本文: 蓝秀文, 陈伟, 高健, 等. 储备池算法学习过程与稳定性分析[J]. 河南师范大学学报(自然科学版), 2026, 54(2): 124-134. (Lan Xiuwen, Chen Wei, Gao Jian, et al. Learning process and stability analysis of reservoir computing [J]. Journal of Henan Normal University(Natural Science Edition), 2026, 54(2): 124-134. DOI: 10.16366/j.cnki.1000-2367.2025.02.22.0002.)

本研究提出了一种最简单的单节点储备池,仅包括一个计算神经元,并通过结合带常数项和泄露项构成了最简单的计算单元.研究发现这个最简单的储备池同时具备时间序列的学习能力,以及不同参数下的预测结果的偏移.以 logistic 映射  $x_{t+1} = kx_t(1 - x_t)$  产生的时间序列为例,在相同的训练数据下,不同的储备池参数可以得到不同的预测状态,既有与训练数据相同的,也有发生了偏移的.例如,训练数据为周期信号,预测时发生混沌,或者反之,训练数据为混沌信号,预测时却产生了混沌,如图 1 所示.



注: (a)、(b)、(c)、(d)和(e)分别表示输入周期 I、周期 II、周期 III、周期 IV 和混沌序列,灰色线表示预测的序列.

图1 单节点储备池预测logistic映射时间序列的结果

Fig.1 The result of predicting the logistic map time series with a single-node reservoir computing

单节点储备池构成了一个研究储备池工作和失效机制的极简模型.围绕其动力学,本文分析单节点储备池算法的学习过程以及预测过程,并分析了对应的动力系统的稳定性,得出了参数与最终状态之间的关系,发现训练数据与预测序列时间差异的关键来自储备池自身形成动力系统的稳定性,并在多节点的经典储备池算法中进行了验证.

本文结构如下:第 1 部分介绍了本文提出的单节点储备池算法;第 2 部分系统地描述并分析了单节点储备池学习过程,并得到了稳定性对于学习结果的影响;第 3 部分将单节点储备池的结果在多节点算法中进行了验证,并在最后第四部分对全文做了总结.

## 1 单节点储备池算法

单节点储备池的示意图如图 2 所示.在训练阶段,储备池计算节点的状态

$$r_{t+1} = (1 - \alpha)r_t + \alpha \tanh(\omega_{in} u_t + b),$$

其中,  $\alpha$  是储备池系统动态的自然速率,  $b$  是偏置项,用于打破储备池内在的对称性.需要注意的是,输入权重  $\omega_{in}$ ,  $\alpha$  和  $b$  在训练阶段之前随机生成,并且在训练过程中保持不变.

对于单节点的储备池,仅靠一维时间序列  $r$  自身很难完成后续学习任务,为此,在其基础

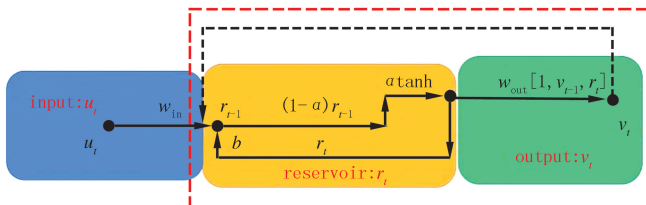


图2 单节点储备池系统训练阶段的结构图

Fig.2 Structural diagram of the training phase of the single-node reservoir computing

上增加了常数 1 以及输入序列  $\mathbf{u}$ , 合并为一个三维的时间序列进行训练. 取储备池层的时间序列矩阵

$$\mathbf{R} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ u_1 & u_2 & \cdots & u_{n_t} \\ r_2 & r_3 & \cdots & r_{n_t+1} \end{bmatrix},$$

其中,  $n_t$  是输入时间序列的长度. 包含所有输出值的输出向量  $\mathbf{v} = \mathbf{W}_{\text{out}} \cdot \mathbf{R}$ . 优化输出权重向量  $\mathbf{W}_{\text{out}} = \arg\min(|\mathbf{W}_{\text{out}} \cdot \mathbf{R} - \mathbf{u}|^2)$ . 向量  $\mathbf{u}$  包含所有输入值. 根据岭回归算法,  $\mathbf{W}_{\text{out}} = (\mathbf{u}\mathbf{R}^T)(\mathbf{R}\mathbf{R}^T + \epsilon\mathbf{E})^{-1}$  其中,  $\mathbf{E}$  是单位矩阵,  $\mathbf{R}^T$  是时间序列矩阵的转置,  $\epsilon$  是一个小的正数, 用于防止算法的过拟合. 选定  $\epsilon = 10^{-10}$ .

在训练阶段获得适当的  $\mathbf{W}_{\text{out}}$  后, 就可以通过储备池算法进行时间序列预测. 在预测阶段, 通过自反馈闭环可以得到预测的时间序列  $v_t$

$$\begin{cases} r_{t+1} = (1 - \alpha)r_t + \alpha \tanh(\omega_{\text{in}}v_t + b), \\ v_{t+1} = c_1 + c_2v_t + c_3v_{t+1}. \end{cases} \quad (1)$$

对于储备池算法的预测效果, 通过将预测的时间序列  $\mathbf{v}$  与实际时间序列  $\mathbf{u}$  进行比较, 并使用预测长度  $L$  定量估计训练后的储备池性能:

$$\begin{cases} |v_i - u_i| \leq \zeta, i = 2, \dots, L, \\ |v_{L+1} - u_{L+1}| > \zeta, \end{cases}$$

其中,  $\zeta$  是截断误差, 在所有后续计算中设  $\zeta = 0.01$ . 显然,  $L$  越大, 储备池系统的学习效果越好.

## 2 单节点储备池算法的学习结果分析

注意到储备池学习过程中, 关键的训练环节基于的是时间序列  $\mathbf{v}$  和  $\mathbf{u}$  之间的拟合. 假设当单节点储备池算法能够完全学习给定输入时间序列的动态特性时, 预测的时间序列  $\mathbf{v}$  应与输入时间序列  $\mathbf{u}$  严格一致. 根据方程(1), 可以得到

$$\begin{cases} r_{t+1} = (1 - \alpha)r_t + \alpha \tanh(\omega_{\text{in}}u_t + b), \\ u_{t+1} = c_1 + c_2u_t + c_3r_{t+1}, \end{cases} \quad (2)$$

如果存在满足方程(2)的输出权重  $\mathbf{W}_{\text{out}} = [c_1, c_2, c_3]$ , 则单节点储备池可以准确地学习给定的输入时间序列. 否则, 单节点储备池的预测时间序列只能在一定步长内与输入序列一致. 当输入序列的周期长度小于或等于 3 时, 方程(2)是欠定的或良定的. 在这种情况下, 满足该方程的输出权重是存在的, 因此单节点储备池输出序列可以严格重现输入周期 I、周期 II 或周期 III 序列. 对于更高周期的序列或混沌序列, 方程(2)将变成一个超定方程. 在此时, 一般时间序列中没有输出权重  $\mathbf{W}_{\text{out}} = [c_1, c_2, c_3]$  能满足方程(2).

根据对方程(2)的分析, 可以发现以周期 III 为分界, 周期更短的序列可以通过方程(2)完全匹配, 而周期更长的序列则无法保证方程(2)的满足. 由此产生的直接的推论是, 单节点储备池算法可以完全学习周期 I、周期 II 或周期 III 序列, 但无法实现周期 IV 和混沌序列的学习. 但根据图 1 中展示的结果, 这个推论并不正确. 一方面, 在学习周期 II 或周期 III 序列时, 仍然会出现学习失败的情况, 预测序列和训练序列不一致. 另一方面, 储备池在合适参数下, 仍然可以实现周期 IV 和混沌序列的学习. 针对这两个问题, 对每一个具体序列的学习情况进行分析.

### 2.1 周期 I 的学习结果

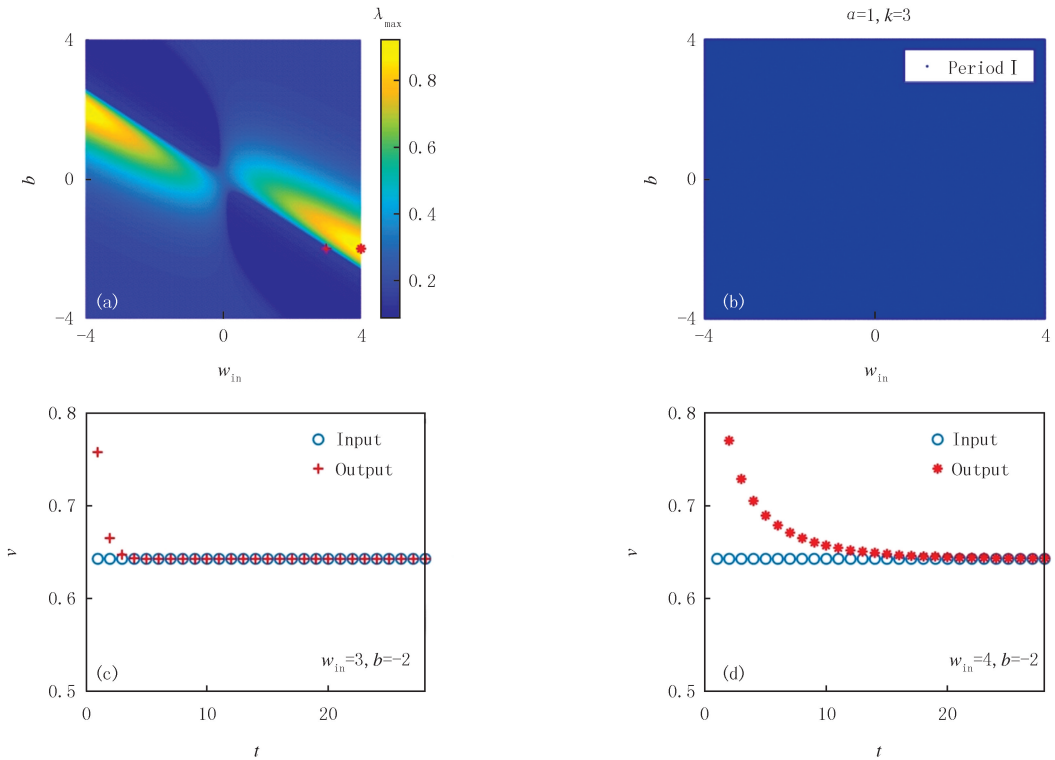
首先, 分析最简单的情况, 即当输入时间序列为周期 I 序列时, 单节点储备池算法的学习效果. 如图 1(a) 所示, 在这种情况下, 无论如何改变参数, 预测序列始终是与输入序列完全一致的周期 I 序列. 此时周期 I 的解为  $r_t = r^*$ ,  $v_t = u_1$ . 将此条件代入方程(1)得到

$$\begin{cases} r^* = \tanh(\omega_{\text{in}}u_1 + b), \\ v^* = u_1. \end{cases}$$

这个解是方程(1)的不动点. 根据预测阶段储备池的动力学, 在  $r_t = r^*$ ,  $v_t = v^*$  处的雅可比矩阵为  $\mathbf{J} =$

$\begin{bmatrix} 1-\alpha & \alpha w_{in} S \\ c_3 & c_2 \end{bmatrix}$ , 其中,  $S = 1 - \tanh^2(w_{in} v^* + b)$ . 雅可比矩阵的 2 个特征值为  $\lambda = ((c_2 + 1 - \alpha) \pm \sqrt{(c_2 + 1 - \alpha)^2 + 4c_3 \alpha w_{in} S})/2$ , 最大特征值的记作  $\lambda_{max}$ . 从图 3(a) 可以看出, 无论如何改变参数  $w_{in}$  和  $b$ , 雅可比矩阵的特征值的绝对值始终小于 1. 这表明在此参数范围内解  $v^*$  是稳定的. 储备池可以输出稳定的此周期 I 时间序列(图 3(b)).

但要注意的是, 此时不同参数的储备池的动力学也仍然是不同的. 尽管它们都是稳定的周期 I 序列, 但由于不同的特征值  $\lambda_{max}$ , 在受到干扰后收敛到固定点的速度不同. 图 3(c) 和 (d) 显示了在相应参数下特征值  $\lambda_{max}$  的演变, 分别接近 0.1 和 0.9. 即便对于简单的周期 I 的解, 储备池算法的动力学仍然会受到其预设参数  $w_{in}$  和  $b$  的影响. 只是此时的影响较小, 无法改变其预测时输出的稳定状态. 储备池的输出预测结果是训练数据与储备池动力学的联合作用.



(a) 特征值  $\lambda_{max}$  的相图; (b) 预测序列相图, 蓝色区域对应周期 I 序列; (c)、(d) 预测序列的结果, RC 参数分别为  $a=1, w_{in}=3, b=-2$  和  $a=1, w_{in}=4, b=-2$ .

图3 周期 I 序列储备池学习结果

Fig.3 Learning results of the reservoir computing for the period I sequence

### 2.2 周期 II 的学习结果

储备池本身的动力学对于预测的影响, 在周期 I 中并不明显. 为此, 考虑更加复杂的时间序列, 当单节点储备池学习 logistic 映射的周期 II 时间序列时, 如图 1(b0) 所示, 预测序列存在 2 种结果. 第 1 种是图 1(b1) 所示的周期 I 时间序列, 它与输入时间序列不一致. 第 2 种是与输入时间序列高度一致的周期 II 时间序列, 如图 1(b2) 所示. 此时不仅仅是收敛速率, 系统的稳定解也发生了改变.

对于学习周期 II 序列, 方程(1)写作:

$$\begin{cases} r_2 = (1 - \alpha)r_1 + \alpha \tanh(w_{in} v_1 + b), v_2 = c_1 + c_2 v_1 + c_3 r_2, \\ r_1 = (1 - \alpha)r_2 + \alpha \tanh(w_{in} v_2 + b), v_1 = c_1 + c_2 v_2 + c_3 r_1. \end{cases} \quad (3)$$

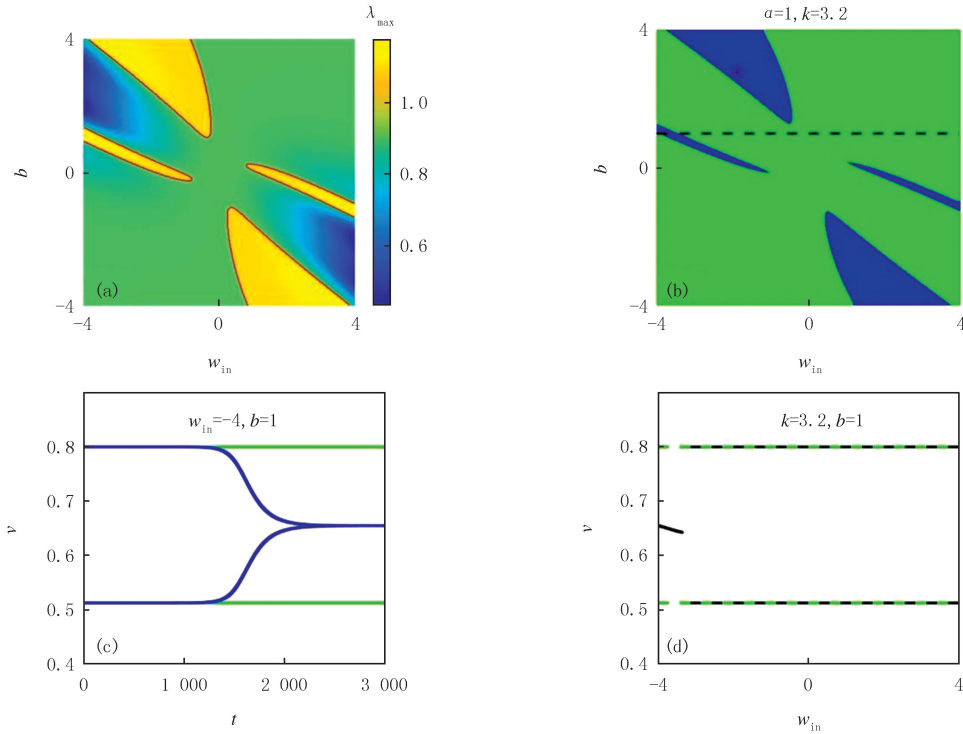
周期 II 的解可以从中解得:

$$\begin{cases} r_1^* = \frac{(1-\alpha)\tanh(\omega_{in}u_1+b) + \tanh(\omega_{in}u_2+b)}{2-\alpha}, v_1^* = u_1, \\ r_2^* = \frac{\tanh(\omega_{in}u_1+b) + (1-\alpha)\tanh(\omega_{in}u_2+b)}{2-\alpha}, v_2^* = u_2. \end{cases} \quad (4)$$

这个解同样是储备池动力学的周期 II 的定解.对于其稳定性,储备池动力学,方程(1)在  $r_1=r_1^*, r_2=r_2^*, u_1=v_1^*, u_2=v_2^*$  处的雅可比矩阵

$$J = \begin{bmatrix} 1-\alpha & 0 & \alpha\omega_{in}s_1 & 0 \\ 0 & 1-\alpha & 0 & \alpha\omega_{in}s_1 \\ 0 & c_3 & c_2 & 0 \\ c_3 & 0 & 0 & c_2 \end{bmatrix},$$

其中,  $s_1=1-\tanh^2(\omega_{in}v_1^*+b), s_2=1-\tanh^2(\omega_{in}v_2^*+b)$ .根据  $J$ ,可以得到雅可比矩阵的 4 个特征值,并取最大特征值记作  $\lambda_{max}$ .当  $|\lambda_{max}| < 1$  时,储备池学到的周期 II 的解是稳定的,反之,是不稳定的.在图 4(a) 和 4(b) 中,分别展示了  $\lambda_{max}$  和最终稳定状态的序列.可以发现两者是一一对应的.



(a) 特征值  $\lambda_{max}$  的相图,图中实线对应  $|\lambda_{max}|=1$ ; (b) 预测序列相图,蓝色与绿色区域分别对应周期 I 和周期 II; (c) 储备池预测结果,参数为  $w_{in}=-4, b=1$ ; (d) 预测序列的分岔图,参数为  $b=1$ ,对应(b)中的黑色虚线.

图4 周期II序列储备池学习结果

Fig.4 Learning results of the reservoir computing for the period II sequence

有趣的是,当周期 II 的解失稳时总是导向了周期 I 的状态.如图 4(c) 所示,时间序列在前 1 200 步中与输入的周期 II 时间序列基本一致后,但随着后续迭代,预测时间序列逐渐演变为周期 I.同时,随着参数  $w_{in}$  的变化,在参数空间内也出现了对应的周期 I 与 II 的分岔,如图 4(d) 所示.

经过训练,储备池学习后得到的储备池动力学,使得训练序列成为其定解,但仅依赖最小均方误差的学习算法,无法保证定解的稳定性.在简单的周期 I 情况下,定解保持稳定,只是收敛速率受到储备池参数影响,但在稍复杂的周期 II 的情况下,定解的稳定性也无法保障,储备池的预测结果会在任意微小的误差或者扰动的情况下,快速偏移定解,产生了预测序列和输入序列的大幅偏差.

注意到方程(4)是欠定的,图 4 使用岭回归方法来获得唯一的  $W_{out}=[c_1, c_2, c_3]$ ,但由于其欠定性,可以得到其他的可能解.在图 5 中,通过分别固定  $c_2=1, c_2=4$ ,可以发现更加丰富分岔特性,除了周期 I 与 II 之

外,还出现了周期 III, IV 乃至混沌的序列.通过分岔图,可以发现,此时的储备池系统动力学可以根据参数的选择不同,展示出完整的倍周期分岔过程.而在整个分岔过程中,训练数据对应的周期 II 均为系统的定解,如图 5(e)和(f)所展示的.

在欠定状态下,不同的额外约束条件,对应着不同解集的选择,也对应着不同的相图.对比图 4(a)、图 5(b)和图 5(d),目标的周期 II 时间序列的稳定区间也随着约束条件的不同而改变.在以上的例子中, $c_2=1$  是比  $c_2=4$  或者岭回归算法更加好的额外条件.所有的解集存在共同的属性,也就是训练集作为定解横亘在整个解集上,但其稳定性区域则受到额外条件的影响.储备池算法具有很强的可塑性,在众多的解集的选择中,岭回归算法并非最优的,但如何选择合适的约束条件,仍然是一个开放的问题.

### 2.3 周期 III 的学习结果

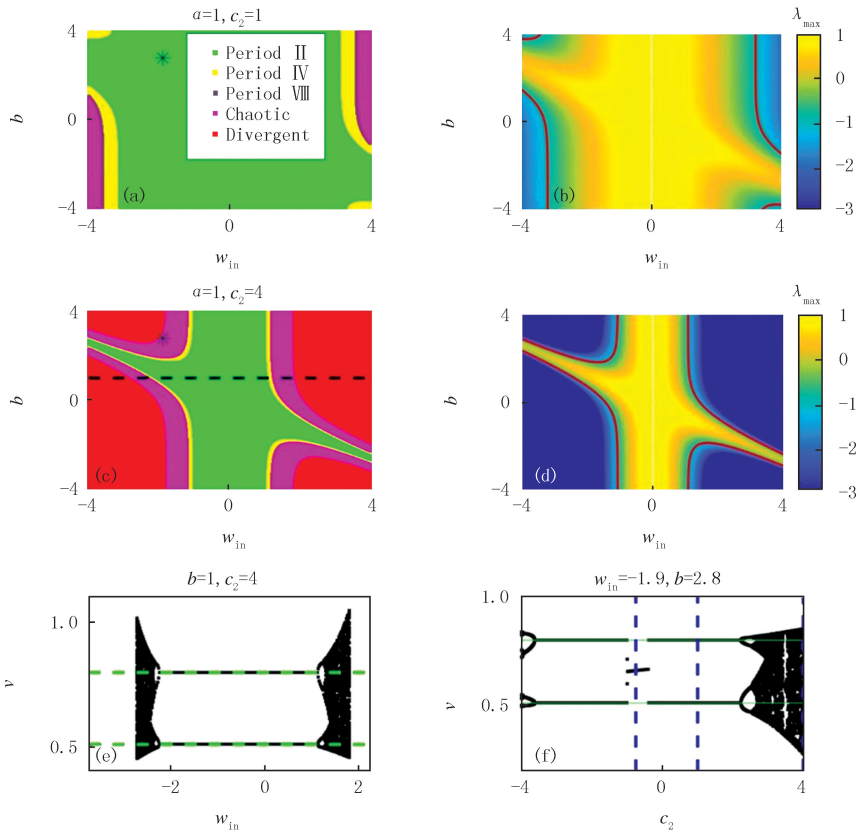
储备池预测序列与训练序列之间存在很深的联系.在 logistic 的周期序列中,周期 I、II、IV 依次通过倍周期分岔产生,而周期 III 则通过切分岔,在混沌区中产生.当一个单节点储备池学习 logistic 映射的周期 III 时间序列时,如图 1(c0-c5)所示.可以学习到的预测序列更加多样,不仅包括周期 I、周期 II 和周期 III 的状态序列,还包括周期 IV 和混沌序列.

周期 III 的学习结果与储备池稳定的分析与周期 II 的分析方法一致.由方程得到对应的 6 阶雅可比矩阵,进而得到相应的 6 个特征值.记录较大特征值为  $\lambda_{\max}$ ,并在图 6(a)中显示了  $\lambda_{\max}$  随参数  $w_{in}$  和  $b$  变化的相图.图 6(b)显示了实际仿真结果比较图.可以看出,在图 6(a)中,  $|\lambda_{\max}| < 1$  的参数区域,使得周期 III 定解稳定,完全与图 6(b)中青绿色区域一致.而由于周期 III 与混沌的关系,在青色的周期 III 区域两侧,都出现了混沌区域.这与 logistic 映射的分岔性质是一致的.对比周期 III 与周期 II 的相图,可以发现储备池学习的结果与序列本身在原动力系统中的分岔性质有密切关系.

### 2.4 周期 IV 的学习结果

对于具有较长周期的 logistic 映射序列,如 logistic 映射的周期 IV 序列,将周期  $u_{t+4} = u_t$  条件代入方程(2),可以得到一个超定方程:

$$\begin{cases} r_2 = (1 - \alpha)r_1 + \alpha \tanh(w_{in}v_1 + b), v_2 = c_1 + c_2v_1 + c_3r_2, \\ r_3 = (1 - \alpha)r_1 + \alpha \tanh(w_{in}v_2 + b), v_3 = c_1 + c_2v_2 + c_3r_3, \\ r_4 = (1 - \alpha)r_1 + \alpha \tanh(w_{in}v_3 + b), v_4 = c_1 + c_2v_3 + c_3r_4, \\ r_1 = (1 - \alpha)r_1 + \alpha \tanh(w_{in}v_4 + b), v_1 = c_1 + c_2v_4 + c_3r_1. \end{cases}$$

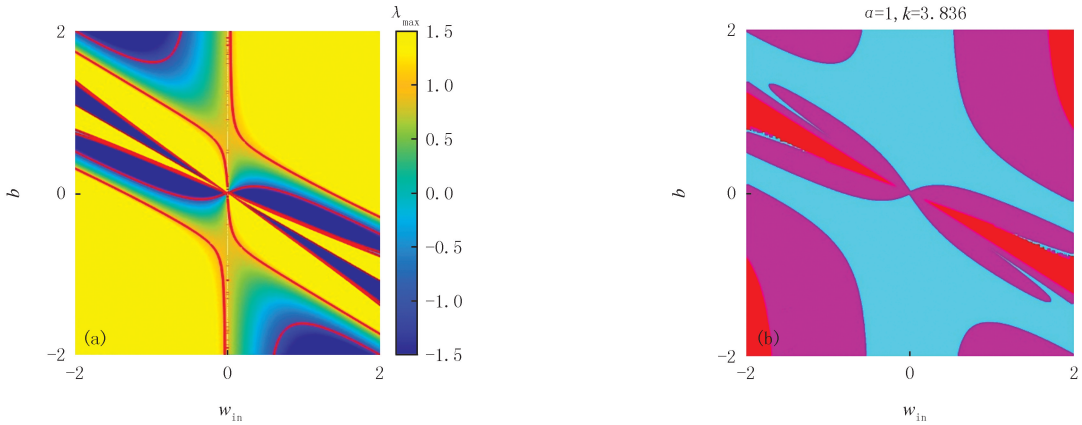


(a) 自反馈演化序列的相图; (b) 当  $c_2=1$  时, 雅可比矩阵最大特征值  $\lambda_{\max}$  随参数  $w_{in}, b$  的变化; (c) 输出权重  $c_2=4$  时的相图; (d) 当  $c_2=4$  时, 最大特征值  $\lambda_{\max}$  随参数  $w_{in}, b$  的变化; (e) 为 (c) 中黑色虚线对应的参数极值图; (f) 固定参数  $w_{in}=-1.9, b=2.8$ , 连续变化  $c_2$ , 预测时间序列的极值图; (f) 中的 3 条蓝色虚线分别对应图 4(b)、图 5(a) 和图 5(c) 中的黑色星号点, 从左到右. (b) (d) 中图中实线对应  $|\lambda_{\max}|=1$ .

图 5 周期 II 序列储备池学习结果分析

Fig.5 Analysis of the learning results of the reservoir computing for the period II sequence

对于一般的  $u_1, u_2, u_3, u_4$ , 无论如何选择参数  $\alpha, w_{in}, b$ , 都无法找到特定的输出权重  $W_{out} = [c_1, c_2, c_3]$  来确保预测时间序列与输入时间序列完全一致. 单节点 RC 无法严格学习并重现 logistic 映射的周期 IV 序列.

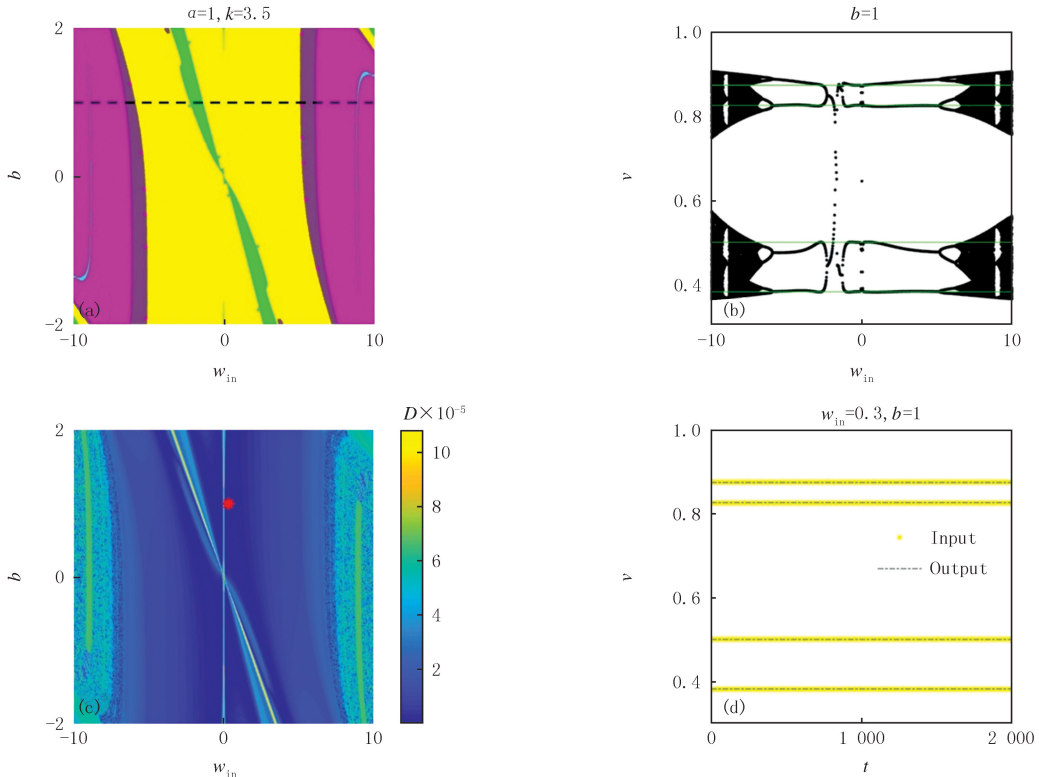


(a) 特征值  $\lambda_{max}$  的相图, 图中实线对应  $|\lambda_{max}|=1$ . (b) 预测序列相图, 青色、粉色和红色区域分别对应周期 III、混沌和发散序列.

图6 周期 III 序列储备池学习结果

Fig.6 Learning results of the reservoir computing for the period III sequence

有趣的是, 如图 1(d1)到(d5)所示, 经过单节点储备池, 算法实际运行的结果显示, 仍然可以通过控制参数, 使得储备池算法预测出近似的周期 IV 序列. 与前面的分析类似, 储备周期 IV 序列的稳定性依赖于参数, 并且会随着参数改变而失稳形成分岔结构, 如图 7(a-b)展示的那样. 而此时不同的一点是, 由于储备池能够训练的参数无法保证重现 logistic 映射的周期 IV 序列, 随着参数的变化, 储备池预测的周期 IV 序列会在失稳前首先变得不准确. 虽然保持着周期 IV 的状态, 但是其具体的数值会发生不小的偏移, 如图 7(b)中分岔图中所示.



(a) 预测序列相图, 在图中序列的颜色采用蓝色、绿色、青色、黄色、紫色和粉色区域分别对应周期 I、周期 II、周期 III、周期 IV、周期 VIII 和混沌序列; (b) 预测序列的周期倍增分叉图, 参数  $b=1$ ; (c) 平均误差  $D$  随参数  $w_{in}$  和  $b$  的变化; (d) 预测结果, 参数为  $w_{in}=0.3, b=1$ .

图7 周期 IV 序列学习结果分析

Fig.7 Learning results of the reservoir computing for the period IV sequence

在稳定性的基础上, 对于周期 IV 的分析需要考虑误差的影响. 为此, 可以定量刻画在训练集上输入时间

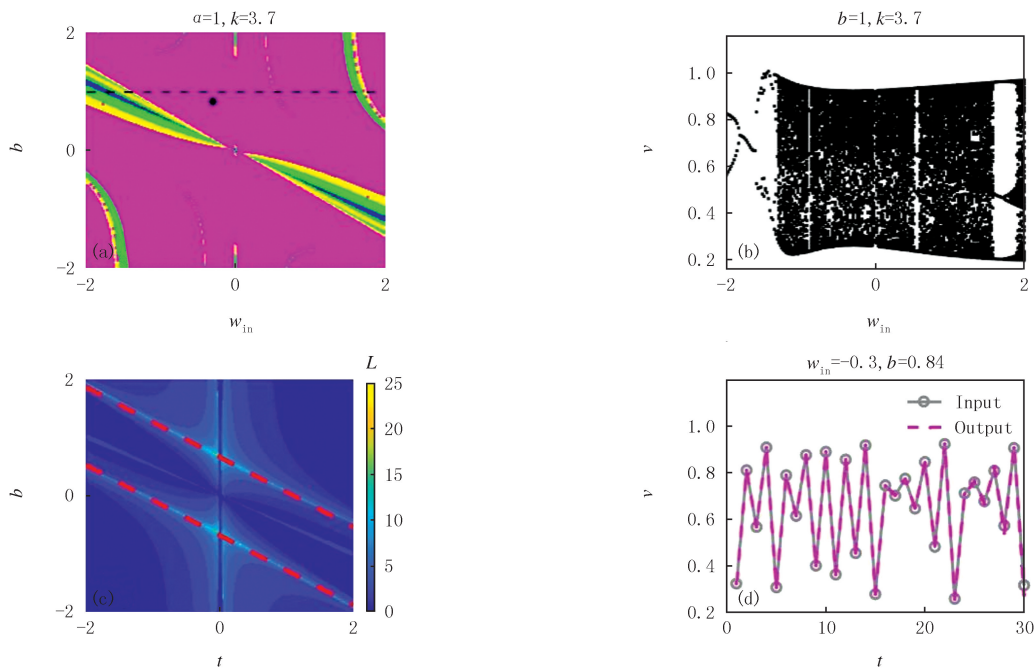
$$D = \frac{1}{n_t} \sum_{t=1}^{n_t} |u_t - v_t|.$$

图 7(c)展示了平均误差  $D$  随参数  $w_{in}$  和  $b$  变化的情况. 可以发现基于稳定性的状态分布图 7(a), 与误差分布图 7(c), 此时出现了大致相同形状. 具有稳定的周期 IV 序列的区域具有更小的误差  $D$ , 而反之误差较大的区域则对应于其他序列. 虽然误差本身的差别不大, 但这种稳定性和误差之间的契合对于仅仅依赖误差进行训练和优化的储备池算法意义重大, 对于它的进一步分析仍然是一个具有挑战的难题.

Logistic 具有更高周期的周期序列, 其结果均与上述结果相似. 随着参数改变, 伴随着稳定性的变化, 预测结果出现大幅偏差, 同时当周期数大于 4 时, 出现了周期状态的不准确. 本文不再对于后续的周期序列状态进行一一分析.

### 2.5 混沌序列的学习结果

最后, 分析混沌序列. 混沌序列可以被视为具有无限周期的序列, 因此无法通过将周期条件代入自反馈方程进行分析. 从图 1(e0) 的仿真结果可以看出, 当输入序列是 logistic 映射的混沌序列时, 预测序列的状态随着参数  $w_{in}, b$  的变化而变化, 包括周期 I、周期 II、周期 IV 和混沌状态, 具体的相图见图 8(a). 除开因为稳定性问题而发生的预测序列的周期性质改变外, 对于混沌状态, 预测仅仅只能发生在有限的几步中.



(a) 预测序列相图, 在图中序列的颜色采用蓝色、绿色、青色、黄色、紫色和粉色区域分别对应周期 I、周期 II、周期 III、周期 IV、周期 VIII 和混沌序列; (b) 预测序列的周期倍增分叉图, 参数  $b=1$ ; (c) 预测步长  $L$  随参数  $w_{in}$  和  $b$  的变化; (d) 预测结果, 储备池参数为,  $w_{in}=-0.3, b=0.84$ .

图8 混沌时间序列学习结果分析

Fig. 8 Learning results of the reservoir computing for the chaotic sequence

虽然混沌轨道的学习和预测非常复杂, 但可以发现, 储备池算法的学习规律仍然是不变的. 图 8(c) 展示了单节点储备池学习 logistic 映射混沌序列时, 随着参数变化, 预测步骤  $L$  的变化情况. 图 8 中的亮点对应于预测效果较好的参数区域, 参数的选择仍然起到关键的作用. 进一步, 可以观察到该区域具有线性特征. 通过拟合线性关系, 可以得到  $b = -0.5w_{in} \pm 0.667$ . 当固定参数  $b=1$  如图 8(a) 中的黑色虚线所示时, 可以看到在  $w_{in}$  变化过程中出现了倍周期分叉, 如图 8(b) 所示. 在选择更好的参数  $w_{in}$  和  $b$  后, 得到的预测时间序列和输入时间序列如图 8(d) 所示. 发现当选择适当的参数时, logistic 映射的混沌时间序列可以在短时间内得到预测.

## 3 从单节点到多节点

在上面的讨论中, 通过对单节点储备池的分析, 发现了其本身的稳定性和对于序列的拟合误差是重要的

两个储备池的性质,而在前人的研究中,稳定性往往被忽略.这是储备池训练与学习中很大一部分意料之外的现象的根本原因.

这一节,进一步讨论节点数量对学习效果的影响,尝试将单节点的储备池分析,尤其是稳定性判断方法,扩展到多节点中.为了方便理论分析,这里考虑在  $\alpha=1$  时学习没有记忆项的周期信号.设输入信号为周期  $m$ , 即  $u=[u_1, u_2, \dots, u_m, u_1, u_2, \dots]$ . 如果 RC 学习周期信号,方程(2)对应的自反馈方程应满足如下方程:

$$\begin{cases} v_1 = c_1 + c_2 v_m + \sum_{i=1}^n c_i + 2 \tanh(\omega_i v_m + b_i), \\ v_2 = c_1 + c_2 v_1 + \sum_{i=1}^n c_i + 2 \tanh(\omega_i v_1 + b_i), \\ \vdots \\ v_m = c_1 + c_2 v_{m-1} + \sum_{i=1}^n c_i + 2 \tanh(\omega_i v_{m-1} + b_i), \end{cases}$$

其中  $\mathbf{v}=\mathbf{u}$ . 方程(1)对应的雅可比矩阵为:

$$\mathbf{J} = \begin{bmatrix} 0 & 0 & \cdots & 0 & S_m \\ S_1 & 0 & \cdots & 0 & 0 \\ 0 & S_2 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & S_{m-1} & 0 \end{bmatrix},$$

其中,  $S_j = c_2 + \sum_{i=1}^n c_i + 2\omega_i [1 - \tanh^2(\omega_i v_j + b_i)]$ . 雅可比矩阵的特征值满足  $\lambda_m = \prod_{j=1}^m S_j$ . 设  $S = |\prod_{j=1}^m S_j|$ , 则当  $S \leq 1$  时,特征值  $|\lambda| \leq 1$ , 此时 RC 输出的周期状态稳定,且该周期状态可以长时间输出. 当  $S > 1$  时,周期状态只能短时间输出.

以 2 个节点的储备池算法为例,分别学习周期 III 和周期 IV 的 logistic 映射,结果如图 9 所示.

假设第一个节点的参数为  $\omega_1=0.22, b_1=1$ , 稳定性指数  $S$  随参数  $\omega_2$  和  $b_2$  变化的情况如图 9(a-b) 所示, 其中  $S > 1$  的不稳定区域用空白表示. 与前面单节点的情况已知,储备池的稳定性对于预测效果的影响,在多节点的情况同样起作用,如图 9(c-d) 所示. 同时,当 2 个节点学习周期 III 状态时,存在大量的不稳定区域,而学习周期 IV 状态时则稳定. 序列本身在原动力系统中的分岔性质对于储备池学习有重要影响,与单节点时所发现的现象一致.

基于图 9(a-b) 的发现,可以通过稳定区域的面积或者比例来刻画节点数的影响. 具体来说,逐步增加节点数量  $n$ , 并且保证当前  $n-1$  个节点的参数  $\omega_i, b_i$  位于稳定区域时,第  $n$  个节点的参数  $\omega_n, b_n$  在区间  $[-2, 2]$  上扫描,以获得对应于  $S \leq 1$  的稳定状态的比例  $P$ . 考虑到随机因素,随机选择  $n-1$  个节点的参数  $\omega_i, b_i$  组合 100 次,得到的均值和方差如图 9(e-f) 所示. 发现随着节点数量的增加,稳定区域的比例  $P$  渐渐接近 1, 这表明当节点数量较大时,随机选择节点参数可以确保稳定性. 对于 logistic 映射和随机选择的储备池,30 个节点足以确保 90% 的成功率来预测正确的状态. 这种节点的互补性对于稳定性的影响,是保证储备池算法工作的基础,但其背后的机制仍然是一个有挑战的开放问题.

## 4 结 论

在本文中,考虑了从单个节点的最简单储备池,以学习 logistic 映射的各种状态为例,对于储备池算法的学习与预测过程进行了分析. 学习过程被抽象为一个非线性动态过程,以研究参数对储备池学习效果的影响. 在仿真中,简单的单节点储备池在学习 logistic 映射时展示了非常丰富的学习结果. 如果选择了合适的参数,单节点储备池可以准确地学习周期 I、周期 II 和周期 III 的 logistic 映射. 对于更复杂的周期 IV 和混沌 logistic 信号,在适当的参数条件下,预测序列也能在近似预测或者短时间内预测输入序列,且误差精度在可接受范围内.

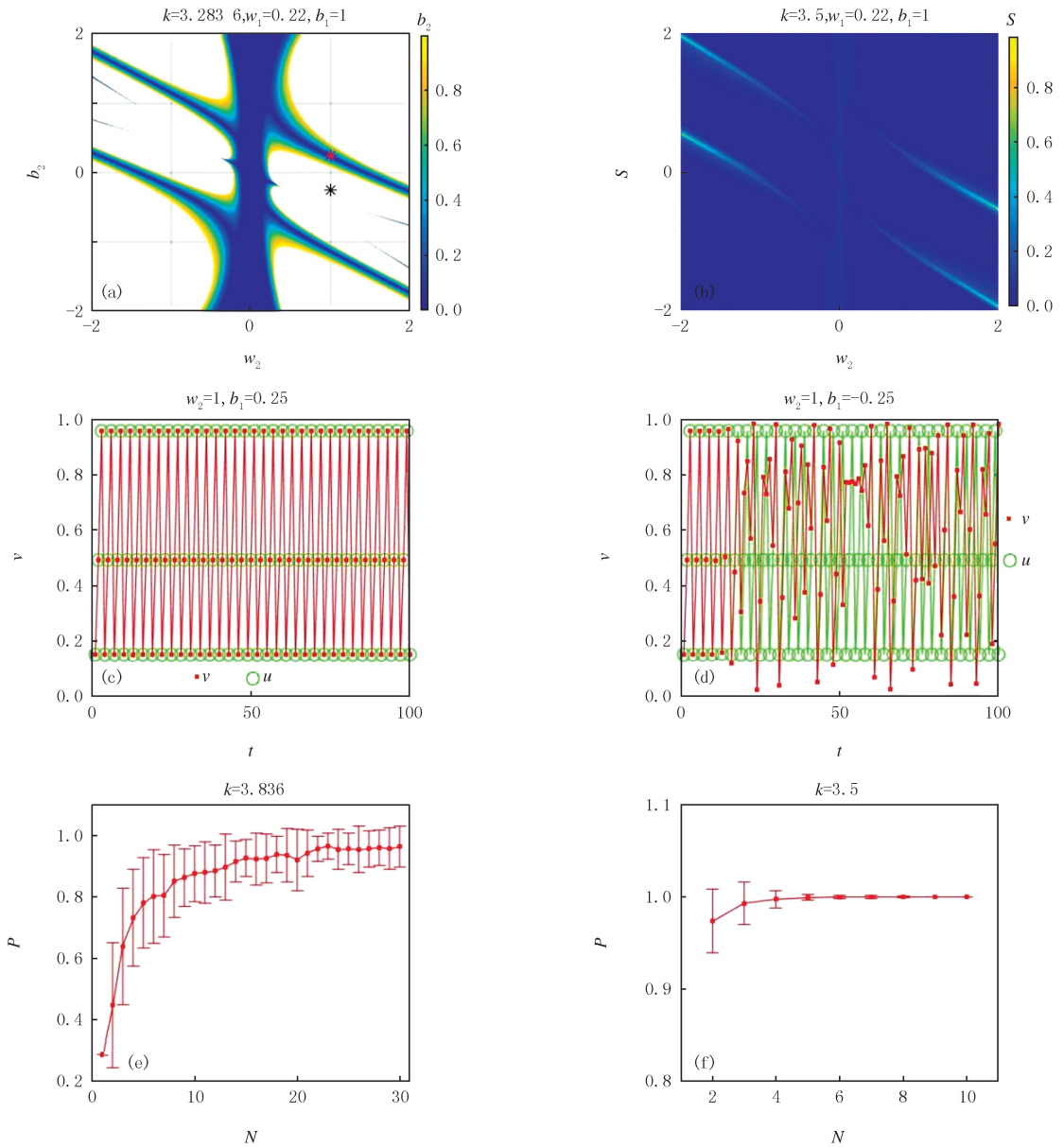


图9 少节点的时间序列分析

Fig.9 Time series analysis for reservoirs with a small number of nodes

基于单节点的分析,本文发现了影响储备池算法预测结果的稳定性因素.在不考虑学习误差的周期 I、周期 II 和周期 III 的情况下,稳定性作为唯一的因素影响预测结果.储备池算法中经常遇到的训练良好但预测发散等情况,都可以在单节点储备池上复现,并通过稳定性进行分析与理解.最后,本文将稳定性分析推广到了多节点的情况,发现其同样适用,并基于此提出了刻画节点数量影响的定量指标,发现随着节点数量的增加,稳定性问题可以被一定程度的改善.这背后的多节点之间的协同作用,可以作为后续研究的重要方向.

### 参 考 文 献

[1] SHUMWAY R H,STOFFER D S,STOFFER D S.Time series analysis and its applications[M].New York: springer,2000.  
 [2] BAILER-JONES C A,MACKAY D J,WITHERS P J.A recurrent neural network for modelling dynamical systems[J].Network,1998,9(4):531-547.  
 [3] CHOI E,SCHUETZ A,STEWART W F,et al.Using recurrent neural network models for early detection of heart failure onset[J].Journal of the American Medical Informatics Association,2017,24(2):361-370.  
 [4] 杨兴耀,肖瑞,卢进堂.新疆维吾尔语口音普通话短文的语音识别研究[J].东北师大学报(自然科学版),2024,56(4):72-80.

- [5] JAEGER H, HAAS H. Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication[J]. *Science*, 2004, 304(5667): 78-80.
- [6] MAASS W, NATSCHLÄGER T, MARKRAM H. Real-time computing without stable states: a new framework for neural computation based on perturbations[J]. *Neural Computation*, 2002, 14(11): 2531-2560.
- [7] LUKOŠEVIČIUS M, JAEGER H. Reservoir computing approaches to recurrent neural network training[J]. *Computer Science Review*, 2009, 3(3): 127-149.
- [8] 高健, 颜子翔, 肖井华. 储备池算法与动力系统分析研究进展[J]. *北京师范大学学报(自然科学版)*, 2023, 59(6): 860-868.  
GAO J, YAN Z X, XIAO J H. Reservoir computing and dynamical systems: a survey[J]. *Journal of Beijing Normal University(Natural Science)*, 2023, 59(6): 860-868.
- [9] 纳治国, 吐尔逊·艾迪力别克, 巴合提古丽·阿斯里别克. 肾上腺轴激素脉冲分泌中的混沌特征研究[J]. *东北师大学报(自然科学版)*, 2024, 56(4): 89-97.
- [10] LU Z X, HUNT B R, OTT E. Attractor reconstruction by machine learning[J]. *Chaos*, 2018, 28(6): 061104.
- [11] DU Y, LI Q, FAN H W, et al. Inferring attracting basins of power system with machine learning[J]. *Physical Review Research*, 2024, 6: 013181.
- [12] LUO H B, DU Y, FAN H W, et al. Reconstructing bifurcation diagrams of chaotic circuits with reservoir computing[J]. *Physical Review E*, 2024, 109: 024210.
- [13] WANG L, FAN H W, WANG Y F, et al. Inferring synchronizability of networked heterogeneous oscillators with machine learning[J]. *Physical Review E*, 2023, 107: 024314.
- [14] ZHANG H, FAN H W, WANG L, et al. Learning Hamiltonian dynamics with reservoir computing[J]. *Physical Review E*, 2021, 104: 024205.
- [15] WANG L, FAN H W, XIAO J H, et al. Criticality in reservoir computer of coupled phase oscillators[J]. *Physical Review E*, 2022, 105: L052201.
- [16] CARROLL T L, PECORA L M. Network structure effects in reservoir computers[J]. *Chaos*, 2019, 29(8): 083130.
- [17] JIA L Y, YAN Z X, ZHAO H, et al. Polynomial function error stair of reservoir computing and its applications in characterizing the learning capability[J]. *Nonlinear Dynamics*, 2024, 112(24): 21751-21764.
- [18] XIE Q Y, YAN Z X, ZHAO H, et al. Large sampling intervals for learning and predicting chaotic systems with reservoir computing[J]. *Journal of Physics A: Mathematical and Theoretical*, 2024, 57(29): 295702.
- [19] LIU S Y, XIAO J H, YAN Z X, et al. Noise resistance of next-generation reservoir computing: a comparative study with high-order correlation computation[J]. *Nonlinear Dynamics*, 2023, 111(15): 14295-14308.

## Learning process and stability analysis of reservoir computing

Lan Xiuwen<sup>1</sup>, Chen Wei<sup>2</sup>, Gao Jian<sup>1</sup>, Yan Zixiang<sup>1</sup>, Xiao Jinghua<sup>1</sup>

(1. School of Science; Laboratory of Mathematics and Information Networks(Ministry of Education), Beijing University of Posts and Telecommunications, Beijing 100876, China; 2. College of Information and Network Engineering, Anhui Science and Technology University, Chuzhou 233100, China)

**Abstract:** Reservoir computing has been widely applied to learning and predicting dynamic behaviors, enabling the prediction of complex temporal signals, including chaotic systems. However, like most neural network algorithms, the learning process of reservoir computing is often considered a black box, lacking systematic research and explanation. This paper analyzes the learning and prediction process as a nonlinear dynamical system by establishing a minimal single-node reservoir computing model. The study found that there is a complex mapping relationship between the training sequence and the output sequence, which is closely related to the dynamical bifurcation properties of the system to be learned. Through stability analysis, the phase diagram of the correspondence between input and output states was obtained, revealing the changes in the stability of the reservoir system associated with prediction failure phenomena. Based on the analysis of the single-node system, it was further discovered that as the size of the reservoir increases, the probability of prediction failure decreases rapidly, and the corresponding stability parameter region of the reservoir expands rapidly. Using the logistic map corresponding to periodic and chaotic signals as an example, the theoretical analysis matches the numerical results completely. By analyzing the learning process and stability of the reservoir, this paper provides a theoretical foundation from a dynamical perspective for the success of algorithms in learning and predicting various dynamic behaviors.

**Keywords:** reservoir computing; machine learning; nonlinear dynamics; stability analysis