

博士論文（要約）

自律的に学習する
高次元力学系の構成論的研究

井上克馬

概要

生物のように適応的かつ頑健に振る舞える知的システムは、高度な自律性と学習機能を兼ね備えている。そのようなシステムはまず外部からの介入が存在し続ける状況下でも常に自発的に活動し、絶え間なく学習しながら振る舞いを獲得し続ける。また学習する対象を外部の環境から自ら選択するのみならず、時にシステムの内的な状態から構成できる。そしてその学習機能は、脳や神経系のようにある複雑な相互作用の中で一貫して実現される。このような「自律的な学習機能」は生物知能の重要な特徴の一つであり、かつ現在の人工知能では十分に達成されていない。

本研究では、脳・身体における多数の要素が相互作用しつつ時間発展する系を抽象化した高次元非線形力学系の上で、自律的な学習機能を実現する数理モデルの構成法を提案する。そして構成された数理モデル、すなわち「自律的な学習システム」の解析を通して自律的な学習機能の背後に通底する普遍的な制約条件を理解することを目論む。特に本研究では自律的な学習機能を部分的に実現した既存の数理モデルとして **Innate Training** に着目する。**Innate Training** は高次元カオス力学系を積極的に活用する構成を採用しているが、そのままでは本研究が目指す自律的な学習システムの要件を完全には満たさない。ここでは上記の点を踏まえ、**Innate Training** を基軸に自律的な学習システムの実現のための4点の改善点を策定し、各々に対して新たな改善手法を提案する。まず **Input Reservoir** と呼ばれる概念を提案し、**Innate Training** で与えられる過渡的入力を高次元非線形力学系内部のダイナミクスとして表現できることを示す(第3章)。この **Input Reservoir** は、学習システムの外部にある機構を部分系として内在化させる着想を与えるため、学習システムの一貫した表現の実現において重要な役割を果たす。次に **Innate Training** では設計者によって制御される入力の生成および切り替えを、**Input Reservoir** ならびに系が生成する高次元カオス軌道を活用し、閉ループを介して高い操作性で設計する手法を提案する(第4章)。この提案手法を駆使することで、設計者からの一定の制御不能性、すなわち独立性が担保された、入力の生成および切り替えの自発的な構成を実現できる。加えて **All-In-One Reservoir** と呼ばれる概念と **Neuralizing** と称される手法を提案し、それらを統合することで、一貫した高次元非線形力学系の中で **Innate Training** の学習メカニズムが内包された **Neuralized All-In-One Reservoir** の具体的な構成法を提案する(第

5 章). この **Neuralized All-In-One Reservoir** は一般的な機械学習モデルとは異なり, 学習部と被学習部が分離しない構成を実現しており, 自己充足的な学習システムの表現に寄与する. さらに, **Innate Training** において設計者によって制御される教師軌道の生成・保持機構そのものが, 系内部のダイナミクスに置換された **Developmental Innate Training** と呼称される構成を提唱する (第 6 章). そして **Input Reservoir** としてカオス力学系が片方向的に結合する構成において, **Developmental Innate Training** が **Innate Training** と同様の学習特性が現れることを示す. さらに **Developmental Innate Training** の解析を通し, 恒常的に学習の進行と崩壊が繰り返される自己組織的臨界現象状のダイナミクスが観測されることを示す. これらの結果は **Input Reservoir** としてカオス的な部分系が内包される構成が恒常的な学習の進行において重要であることを意味し, 自律的な学習機能が要請する系のトポロジーに関する制約の存在を示唆する. 最後に, ここまで提案された **Input Reservoir**, **Neuralizing**, **All-In-One Reservoir**, および **Developmental Innate Training** を統合・駆使し, 自律的な学習機能がニューラルネットワークのダイナミクスの上で実現された「統合学習システム」を構成する (第 7 章). そして得られた統合学習システムが, 本研究において導入された自律的な学習機能の要件を満足する自律的な学習システムであることを示し, その有効性を議論する.

目次

第 1 章	序論	1
1.1	生物知能の自律性	1
1.2	自律的なシステムの構成	2
1.3	人工知能と「自律的な学習機能」	3
1.4	自律的な学習機能の要件	5
1.5	Innate Training と自律的な学習機能	8
1.6	研究の目的	9
1.7	研究の意義	10
1.8	論文の構成	10
第 2 章	Innate Training—その手法と特色	13
2.1	Reservoir Computing とその思想	13
2.2	Echo State Network とその定式化	14
2.3	Readout の学習アルゴリズム	16
2.3.1	Ridge 回帰	16
2.3.2	RLS アルゴリズム	17
2.3.3	FORCE 学習	18
2.4	Innate Training とその特色	18
2.5	自律的な学習システムとしての Innate Training	20
2.6	Innate Training の改善点	21
2.7	自律的な学習システムに向けた拡張の指針	22
2.7.1	過渡的入力の内在化	23
2.7.2	自発的な記号の切り替え	23
2.7.3	学習過程の自己充足的表現	23
2.7.4	教師軌道の自己決定・構成	24
2.7.5	手法の統合および解析	24

第 3 章	Input Reservoir の提案および Innate Training の学習性能への影響	25
3.1	問題設定	25
3.1.1	要求される過渡的入力に関する考察	25
3.1.2	過渡的入力のモデル化および Input Reservoir による内在化	26
3.2	提案手法	27
3.2.1	Chaotic ESN および Innate Training の設定	28
3.2.2	Timer Task および Timer Task Capacity	28
3.3	結果	29
3.3.1	数値計算による評価	29
3.3.2	ソフトボディを用いた実験	30
3.4	議論	32
第 4 章	高次元カオスを活用した自発的な記号の切り替え	35
4.1	問題設定	35
4.1.1	自発的な記号の切り替え	35
4.1.2	系が満たすべき制約	36
4.1.3	関連研究	37
4.1.4	手法の指針	38
4.1.5	カオスの遍歴と自発的な記号遷移	40
4.2	提案手法	40
4.2.1	Chaotic ESN を基盤とする実験系	40
4.2.2	Innate Training による高次元カオス軌道の設計 (Step 1)	42
4.2.3	系自身による周期的な記号の切り替え (Step 2)	43
4.2.4	カオス性を活用した確率的な記号切り替え (Step 3)	43
4.3	結果	43
4.3.1	記号に応じた高次元カオス軌道の安定的生成 (Step 1)	44
4.3.2	周期的な記号遷移の埋め込み (Step 2)	49
4.3.3	確率的な記号遷移の埋め込み (Step 3)	50
4.4	議論	51
4.4.1	提案手法のメカニズムに対する考察	51
4.4.2	提案手法の有効性	54
4.4.3	提案手法の重要性	55
4.4.4	「カオスの遍歴の設計手法」としての提案手法の妥当性	56
4.4.5	自律的な学習システムの構成における提案手法の寄与	57

第 5 章	Neuralized All-In-One Reservoir の提案および高次元力学系への学習過程の埋め込み	59
第 6 章	Developmental Innate Training の提案およびその動的特性	61
6.1	問題設定	61
6.2	提案手法	62
6.2.1	双方向・片方向モチーフ	62
6.2.2	詳細な実験設定	63
6.3	結果	63
6.4	議論	67
第 7 章	統合学習システム–自律的に学習する高次元カオス力学系の構成	71
7.1	問題設定	71
7.2	提案手法	72
7.3	結果	72
7.4	議論	74
第 8 章	結論と展望	79
8.1	結論	79
8.2	今後の展望	80
付録 A	Reservoir Computing の数理的背景	83
付録 B	自発的な記号遷移 補足資料	85
B.1	Innate Training のアルゴリズムおよび詳細設定	85
B.2	Chaotic ESN への投射行列の制御	85
B.3	Innate Training 後の固有値分布の変化	85
B.4	最大 Lyapunov 指数	86
B.5	局所 Lyapunov 指数	87
B.6	長周期の周期的遷移の設計	87
B.7	履歴依存的な確率遷移則の埋め込み	89
B.8	軌道の履歴依存性の評価	89
付録 C	Neuralized All-In-One Reservoir 補足資料	93
付録 D	Developmental Innate Training 補足資料	95
D.1	他条件における解析結果	95

付録 E 統合学習システム 補足資料	97
謝辞	99
参考文献	101

目次

2.1	Reservoir Computing および Innate Training の模式図	15
2.2	Innate Training の学習前後の比較	21
2.3	Innate Training における過渡的入力の重要性	23
3.1	Input Reservoir の概念図	27
3.2	実験系および Timer Task の説明	29
3.3	Innate Training におけるカオス性の重要性の評価	31
3.4	Input Reservoir の性質が与える Innate Training の性能への影響	31
3.5	使用されたソフトボディおよびその制御方法	33
3.6	ソフトセンサを用いた Innate Training の評価	34
4.1	実験系の模式図	39
4.2	Step 1: Innate Training による複数の高次元カオス軌道の埋め込み	46
4.3	Innate Training 後のシステムの性質の解析	47
4.4	Step 2: 周期的遷移の自律的生成	48
4.5	Step 3: 確率的な遷移の自律的生成	52
4.6	記号力学およびその終端状態の解析	53
6.1	Developmental Innate Training の模式図	64
6.2	Developmental Innate Training のデモンストレーション	65
6.3	最大 Lyapunov 指数による解析	67
6.4	ESP Length による解析	68
7.1	統合学習システムの模式図	73
7.2	統合学習システムのデモンストレーション	75
7.3	All-In-One Reservoir を応用した統合学習システムの次元削減	76
B.1	内部結合のスペクトル半径の解析および Timer Task の設定	87

B.2	FORCE 学習と提案手法との設計性の比較	89
B.3	履歴依存的な確率遷移の埋め込み	91
B.4	軌道の履歴依存性の解析	92
D.1	他条件におけるスケールフリー性の解析	96
D.2	他条件における結合行列の解析	96
E.1	次元圧縮された統合学習システムの全ダイナミクス	97

表目次

第 1 章

序論

1.1 生物知能の自律性

生物は高度な自律性を有し，実世界においてしなやかに振る舞える．まず生物は恒常的である．すなわちその自己同一性を維持し，外的な介入から一定の独立性を維持しながら動作できる．また同時に生物は環境の変化に対して適応的である．言い換えれば，時々刻々と変化する状況に対処して，適切な振る舞いを獲得・生成し続けられる．このような自己同一性と適応性の共存^{*1}は，生物知能^{*2}が有する自律性を特徴づける重要な要素である．そしてその理解ならびに実現は人工知能分野における究極的な目標の一つであり，かつ現在の人工知能^{*3}では十分に達成されていない [1]．

一方でこのような生物知能の現象学は，「生命とは何か？」という根源的な問いの形で何世紀にもわたって人々に関心を寄せられてきた．そしてその帰結として生物をある種のシステムと捉え，その全体的な特性を扱うシステム論的なアプローチの確立に至った．システム論以前には，古典的には生命現象の根源をめぐる生気論と機械論の二項対立が生み出された．これらは合目的性をもつ非物質的構成素を認めるか否かの立場によって二分される．近代以降，デカルト的思想を基盤とする要素還元主義は広く成功をおさめ，機械論が生物学的世界を席卷した．この要素還元主義は他方で学問の細分化を生じ，個々の構成要素の説明にはつながっても，現象学が対象とするような生物の全体的な特性の理解や普遍的な法則性の発見には至らなかった．その反省より 1940 年代に生み出されたのが一般システム理論 [2] およびサイバネティクス (Cybernetics) [3] であり，これらはその後のシステム理論の走りとなる．このシステム理論では，非物質的構成素を導入しないという意味で機械論的でありながら，いくつかの要素が相互作用するシステムにおいて現れる全体的な特性が研究の対象となる点で要素還元主

*1 すなわち変化し続けながらも自己が安定的に維持される．

*2 生物の知性，または知的な振る舞い．

*3 知的に振る舞える計算モデルないし人工的な機械．

義と相対する。そしてこのシステム論とその後発展した情報理論 [4] を契機に今日では、生物を自己維持しながら環境との相互作用を通じて自己形成する開放系とみなす視点が広く流布している。実際、散逸構造や動的非平衡系ならびに複雑系と呼ばれる形で、生命現象のモデルとしてこれらは今なお研究され続けている [5, 6, 7, 8, 9].

1.2 自律的なシステムの構成

生物知能を理解するためにシステム論的アプローチが確立されるのと並行して、自己同一性と適応性を兼ね備えた自律的なシステムの構成が試みられてきた。後にサイバネティクスとして知られる領域の成立の過程で Wiener によって理論化されたフィードバック機構は、そのような自律的なシステムを実現する上で必要となる最小原理を提示した最初の例といえよう。フィードバック機構ではシステムの出力が再びシステムに入力される閉ループが構成されることで、外的な介入を必要としない自己制御が達成される。特に生物においては、恒常性 (Homeostasis) を維持する内的なフィードバックと、外環境を介する外的なフィードバックとに分けられ、それぞれ生体内では自律神経系と感覚系に対応する。またこのフィードバック機構は制御工学の根幹となり、この後、制御理論 (現代制御理論, 非線形制御理論等) やニューラルネットワーク制御 [10] と派生してゆく。^{*4}

また生物の重要な特性として自己複製能が挙げられ、生物の恒常性の維持のために積極的な役割を果たしている。この自己複製能を有するシステムの先駆けとして、von Neumann [15] は二次元オートマトン上で自己複製する数理モデルを構成した。この von Neumann のモデルは 200000 セルに及ぶ非常に複雑なものであったが後に、非常に単純な規則を持つライフゲーム (Game of Life) [16] と呼ばれる二次元セルオートマトン上で、自己複製と自己エミュレーションを行うシステムの構成例が示された。一方で一般にこのようなオートマトンはノイズに対して脆弱であることが知られる。Gacs [17] はこのノイズ耐性に関する問題に対処すべく、一次元オートマトン上でどのような外的な摂動に対しても安定的に自己修復するシステムを提案した。これらの自己複製能を有するオートマトンの一連の研究は、生物知能の特性の解明を数理モデルの構成と解析を介して目論む構成論的研究の代表として位置付けられよう。

並行して Maturana と Varela によって提案されたオートポイエーシス (Autopoiesis) [18, 19] は、自律性の要件と自律的なシステムの構成に関して、重要かつ特異的な問題を提起している。オートポイエーシスは生物を含む自己組織的システムの定義であり、以下の4点の特性で特徴づけられる。

^{*4} 昨今脚光を浴びている深層強化学習 [11, 12] もこれらの制御工学と深く関連する。特に強化学習 [13] では一種の適応最適制御問題が扱われているとみなすことができ [14], その一部にニューラルネットワークが導入されたのが深層強化学習の手法である。

1. 自律性：有機構成^{*5}の相互作用の中で，構成素を生成する過程（関係）のネットワークが持続的に再生成される。
2. 個体性：絶えず産出を行い有機構成を不変に保つことにより，観測者とは無関係に同一性が保持される。
3. 境界の自己決定性：その自己産出プロセスの過程で，自らの境界が設定される。
4. 入出力の不在：その構成素の産出の過程において一貫した閉鎖系をなし，かつ環境との関係において外部と内部を区別しない。

オートポイエーシスの先駆的な点，ならびに先述の開放系としての生物の視点との違いは，観測者による外的な記述の存在を強調し，その記述を内在化させる自己言及的なシステムの構成を目指した問題設定にある．生物を開放系としてみなす際，システムと環境との間に横たわる境界は先行的に外部の観測者によって規定される．オートポイエーシスのシステムでは，構成素が構成素を作り出す円環的な過程で作られる閉包領域の実現によって，システムと環境を区別する境界の自動生成を目指している．この過程により結果として，観測者による外的な記述が排除される．また開放系では入出力関係は環境入力に対する応答として解釈されるが，観測者による境界の設定が排除されているため，オートポイエーシスのシステムでは従来の意味での入出力が存在しない．この入出力の不在は結果としてオートポイエーシスの特異性を象徴する特性といえる．^{*6}

1.3 人工知能と「自律的な学習機能」

ここまで生物知能が有する自律性の重要な特徴として自己同一性と適応性を挙げ，その構成を目指した複数のシステム論的アプローチの歴史を辿った．人工知能分野の究極的な目標である，しなやかかつ頑強な生物知能の振る舞いの実装は，これらの研究の延長線上に位置づけられるといえよう．一方で 2022 年現在では，生物知能が有するような高度な自律性を兼ね備えた人工知能の実現には至っていない．本節ではその要因をその学習機能の形態の差異に求め，特に人工知能の中心的技術である機械学習 [22] と生物知能の対比を通し，現在の人工知能に

^{*5} 河本 [19] による“Organization”の訳語であり，同書では『システムを単位体として，システムが経る相互作用や変換の変化を規定する諸関係。』と定義づけられる．またもともとは一般システム理論 [2] の用語であり，個々の要素には還元されない性質を持った構成を指す．オートポイエーシスでは，特に構成素が構成素を生成するプロセスが想定され，そこではシステムはその産出プロセスの有機的なネットワークとして規定される．他方，構成素自身もこの産出プロセスのネットワークそのものを絶えず産出しており，この点において構成素とシステムは産出関係を介して循環的に規定される．

^{*6} なお，このオートポイエーシスは形式的な記述を伴わず議論が進められ，その後も理論の明確化と発展がなされてきたとは言い難い．結果として要素還元主義を徹底させ飛躍的に発展した 20 世紀の生物学に対する影響は限定的なものにとどまり，その価値はむしろシステム論的な斬新さにあったといえよう．また [20] において指摘されている通り，オートポイエーシスの形式化は本質的に矛盾を内包しており，その意味で破綻している．この矛盾の存在をきっかけに，後に内部観測 [21] の発見と開拓に至る．

欠如している要素を考察しよう。

まず学習機能の適応性に対する寄与を論じる。ある環境において生存に有利な形質を生物が有している時、その生物は適応的であると言われる。このような形質は遺伝的なもの、すなわち世代を跨いで獲得される生得的なもの、遺伝的でないもの、すなわちその世代に限定される後天的なもの^{*7}に分けられる。学習機能が寄与するのは後者の意味での適応性、すなわち即応的な立ち振る舞いの獲得と生成においてである。学習はその語義の通り、外部環境との相互作用を通して新たに知識や技能、価値基準等を獲得する営みである。したがって、学習機能を有するシステムは本質的に可塑的であり、その内部状態は変化し続ける。そして学習し続けることでシステムは適応的であることができ、結果的に知的に見える振る舞いが実現される。実際良く知られているように、特にサルやヒト等の霊長類やカラスにおいて特に学習機能が発達しており、その知性を特徴づける重要な特徴である。^{*8}

このような学習機能の人工的な計算モデル上への構成を目論むのが機械学習 [22] の諸手法である。機械学習は、与えられたデータを基に自動で改善するアルゴリズム技術の総称である。そして究極的には機械学習は、人間に匹敵する学習機能を兼ね備えた認知モデル、すなわち人工知能の構成を目指している。近年、計算機性能の向上およびアルゴリズムの改良にともない、機械学習技術が急速に発展している。タスクを限定すれば、人間に匹敵・超越する計算モデルが機械学習によって実現されており、今後その範囲はさらに拡大されるだろう。^{*9}

しかしながらその学習機能の自律性に関して、生物知能と人工知能との間には依然として大きな乖離がある。端的に言えば、現状の機械学習技術によって実現される人工知能は「勝手に」は学習することができず、その挙動の多くは常に設計者の手に委ねられている。^{*10}また機械学習では、しばしばある限られた範囲のタスクのみが想定され汎用的でない。反面、生物知能の学習機能は恒常性を有し、環境との相互作用の中で常に学習し続ける。そして、想定されていない様々な未知の状況に対しても、その学習機能を駆使して適応的な振る舞いを生み出すことができる。このような生物知能と人工知能の現状の差異は、いわゆる「強い AI」と「弱い AI」の違い [27] に換言できるだろう。そして人間に比肩する学習機能を実現するためには、少なくとも人工知能が設計者の手から離れても学習し続けかつ汎用的でなければならない。^{*11}

^{*7} 後天性を強調するため、しばしば順応性と言い換えられる。

^{*8} 無論、身体性認知科学 [1] の文脈で議論されているように、学習機能の有無ないしは多寡のみで知性の度合いを論じることは難しい。Braitenberg [23] によって提案された Vehicle はその極端な「反例」で、学習機能を有せずとも知的に振る舞えるエージェントを構成している。一方で人間やカラスの学習機能は他の生物から一線を画するものであり、その卓越した適応性、すなわち知性に寄与しているといっても過言ではないだろう。

^{*9} 2022 年現在、著名な成果としては、画像認識タスク [24]、囲碁・将棋等の完全情報ゲーム [25]、ポーカー等の不完全情報ゲーム [26] が挙げられる。今後も様々なタスク、すなわちニッチが開拓され続けるだろう。

^{*10} 言い換えれば、せいぜい推論モデル内の各パラメータの決定がアルゴリズムによって「自動で」計算されるのみで、多くの外的な機構が先行的に導入される。例えば、学習率やアルゴリズムといった学習に関する設定（ハイパーパラメータ）の選定は設計者に委ねられる。また学習データの取捨選択や整形、学習の進行・停止の決定は設計者によって制御される。

^{*11} 「弱い AI」の構成、すなわち特定の問題解決に限定されるならば、自律性・汎用性に関する優先度は低い。

また学習機能に関する生物知能と人工知能の大きな相違点として、その構成の自己充足性が挙げられる。言い換えれば生物知能は物理的実体を有し、その構成はそれ単体で独立して機能しているのに対し、人工知能では多くの「付属品」が学習機能を成り立たせるために必要となる。^{*12} 例えば人間を考えた時、その学習機能はある物理的実体（脳、感覚器、筋肉等）とその相互作用の上で実現されており、それらは物理的に制約される。^{*13} 結果的にその学習機能は身体・環境・脳の複雑かつ密な^{*14}相互作用の中で実現されており、総体としてそれ単体で機能する構成をなす。一方で既存の機械学習では、学習機能が仮想的な計算機上のアルゴリズムとして実装される。厳密には CPU（Central Processing Unit）やメモリといった計算機内の物理的実体の上で構成されるが、その身体性は非常に限定的である。^{*15} そして仮想的であるがゆえに設計者による無定形な構成の変更が許され、ここでは生物知能で求められる自己充足性が問題とならない。結果的に機械学習の学習機能は多くの独立した機構の恣意的な組み合わせで成立し、各機構はそれ単体では学習機能を有しない。^{*16}

このように学習機能に関して、生物知能と人工知能の間で比較した時、自律性に関して大きな乖離を生み出しているといえる。生物知能は自己充足的な構成を持ち汎用的で恒常的に学習し続けるのに対し、既存の人工知能は対照的である。したがって、「強い AI」を目指す上で、生物知能に倣い自律性が考慮された「自律的な学習機能」を有する計算モデルの実現が、機械学習の喫緊の課題となるだろう。

1.4 自律的な学習機能の要件

ここまで生物知能が有する自律性の重要な特徴として自己同一性と適応性を論じた。そして既存の人工知能において自律的な学習機能が実現されておらず、結果的に生物知能が有するようないしなやかな振る舞いが達成されていないことを指摘した。本研究ではシステム論アプローチの延長として、自律的な学習機能を兼ね備えた数理モデルである「自律的な学習システム」の構成と理解を課題に設定する。そして本節では、自律的な学習システムが満たすべき要件をこれまでの議論を基に論じる。

実際機械学習ではしばしば、アルゴリズムの取束性やデータ効率性といった学習の「素早さ」が主題となる。逆に言えば、自律性に関する問題は「強い AI」の構成を目指した時にはじめて露呈する問題である。例えば Catastrophic forgetting [28] はその代表的な現象で、既存の深層学習モデルの学習の継続（すなわち恒常的な変化能）と、学習された機能の安定性との両立が困難であることを指摘している。

^{*12} オートポイエシスの言葉を借りれば、このような状況はアロポイエシス（Allopoiesis）的である。

^{*13} 限られたサイズの空間内に配置され、かつその上の物理法則に従っている。

^{*14} 簡単には分離できない。

^{*15} 例えば実数の bit 表現による桁落ち [20] 等の計算機特有の制約があるが、それらが積極的に関与するケースは限られる。

^{*16} 例えば、機械学習モデルを考えた時、モデルの他に多くの独立した機構（パラメータを調整する学習器、学習データを収集・選択する機構、学習の成否を判別する評価関数等）が別々に存在し、設計者によって外的に組み合わされ制御される。

前節で議論されたように、生物知能は高度な自律性を有しその構成は自己充足的である。それらを踏まえ、ここでは自律的な学習システムが満たすべき要件として以下に挙げられる3点に着目する。

要件1 自発性

要件2 学習対象の自己決定・構成能

要件3 学習メカニズムの内包

まず自律的な学習システムは自発的、すなわち外部からの介入があっても完全に従わず、かつ自身の内的なルールに従い学習し続けなければならない。換言すれば自律的な学習システムでは、その内的な機構によって恒常的に学習が進行することが望ましい。この要件は、先述の自己同一性と適応性の両方に対応する。すなわち自発的な学習システムは、常に学習し変化し続けられるという点で適応的であるが、その学習機能は恒常的に保たれている。生物はそのような自発的なシステムである。そして生物はしばしば自身に内包されるカオス性を活用することで、その自発性を実現している。^{*17}例えば脳や神経系はマイクロからマクロに至るあらゆる階層で自発的な活動が現れるカオス的なシステムである [29, 30, 31, 32, 33, 34, 35]。良く知られているように神経細胞は、無入力あるいは単調な入力に対しても複雑かつ予測不能な自発発火のパターンを示す [36, 37]。また集合的な神経活動も外部の状態だけでなく、内的な活動状態の履歴に従って変化することが知られる [38, 39]。^{*18}このように生物は様々なスケールで自発的なシステムであり、その学習機能は自発性を兼ね備えている。対照的に現状の人工知能は自発的ではない。特に機械学習において学習の制御は設計者の手に完全に委ねられている。典型的な設定では、学習モデルのパラメータは「学習フェーズ」のみにおいて調整され、「評価フェーズ」においては調整されない。またカオスの予測不能性は通常設計者にとって手に負えない厄介な要素で、カオスそのものを抑圧する手法は数多く提案されている [41] が、内在的なカオス性を活用する機械学習システムはほとんど存在しない。結果的に、カオス性のような自発的な要素を本質的に活用する生物とは異なり、自発性を内包し「勝手に」動作し続ける手法は機械学習においてほとんど提案されていない。^{*19*20}

^{*17} ここであくまでカオス性は自発性を実現する一手法であることに注意されたい。例えば、量子力学における系の不確定性は、外部から制御できない独立な要素である。

^{*18} 他にもラットの海馬の神経活動の研究において、特定の場所の通過イベントに応じて発火する場所細胞の受容野が、外部環境の変化がない状況でも内的な状態の変化（課題の違い等）によって再配置される現象が観察された [40]。このような再配置は自発的でない学習システムでは起こりえない。

^{*19} なお我々の研究 [42] では学習済みの自然言語処理モデルである ALBERT (A-Lite Bidirectional Encoder Representations from Transformers) [43] で、その過渡ダイナミクスのみでカオス性を示す Transient Chaos [44] が生成されることを示した。ALBERT では予めカオス性が担保されるような初期値は採用してはいないものの、結果としてカオス的な軌道が生成され、かつそれが活用される形で系が学習されるのは非常に興味深い。

^{*20} また外部からの制御不能性という意味ではノイズ、すなわち確率論的な要素を組み込む機械学習手法は多く存在するが、この確率論的な要素は系、特に力学系の時間発展のメカニズムとは分離して存在する機構で、設計

次に、自律的な学習システムは学習対象を自ら決定し、時に構成できなければならない。これは適応性に関連する要請である。すなわち時々刻々と変化する環境を前に、系自身が自ら何を学習すべきかを取捨選択し、状況に応じた臨機応変な振る舞いを獲得することが求められる。生物はそのような学習対象の自己決定・構成能を有する自律的な学習システムである。例えば日常的に我々は、我々自身の内発的な興味に基づいて、何を学習するかを決定している。この時学習対象は外的でも、その選択は内的でありシステムの外部からは完全には根拠づけられない。また生物は環境入力がない状況でも学習対象を構築し学習を行う。実際脳は睡眠中でも覚醒時に経験した記憶を想起し固定化することが指摘されている [45, 46]。このような状況では、学習対象が外部の環境情報ではなく、むしろ系の内的な状態より構成されていると解釈されるほうが自然であろう。反対に、機械学習では一般的に学習データを設計者が予め用意しなければならない。言い換えれば、学習される範囲や対象を先行的に外部の設計者が設定しており、その意味で設計者に依存している。^{*21}

最後に自律的な学習システムは、学習メカニズムをそれ自身の中に内包しなければならない。換言すれば、自律的な学習システムは、その学習を実現する機構に関して自己充足的であることが求められる。この点も機械学習との対比で明確化される。例えば機械学習では学習モデルのパラメータを学習アルゴリズムが調整する、被学習器/学習器 (Optimizee/Optimizer) からなる分離構成が採用される。これは、学習モデルの他に学習器の機構が別途設置され、推論を行う被学習器単体では学習できないことを意味する。対照的に生物は、この機械学習ほど明示的に分離された構成を持たず、ある物理的実体そのものだけで学習機能を成立させることができる。言い換えればある一貫したシステムの内部状態の動的な変化、すなわちダイナミクスの上でその学習機能が実現されていると解釈できる。^{*22}

以上をまとめれば自律的な学習システムでは、自発的に学習が進行され、そのシステムの相互作用の中で学習対象の選択・構成がなされ、かつ学習メカニズムそのものが一貫して表現され、それ単体で学習機能が実現されることが望ましい。本研究では、そのような自律的な学習機能を備えた数理モデルの設計論を構築する。具体的には、脳・身体における多数の要素が相互作用しつつ時間発展する系を抽象化した高次元非線形力学系^{*23}の上で、自律的な学習機能を

者によって先行的に付与されるものである。これはこの後に議論される要件3の自己充足性の問題と関連する。
^{*21} ここで注意したいのは、学習システムを想定した時、必然的に環境とシステムとの境界が導入されているという点である。すなわち、オートポイエーシスで議論されたような境界の自己決定性はここでは扱われない。結果的に、本研究で扱われる自律的な学習機能は外界との相互作用で特徴づけられ、常に環境からの受動性を孕む。これは第7章においても議論される。

^{*22} 無論、生物でも性質の異なる要素（例えば神経細胞における化学的・電気的反応）を同一システム内に含み、それらは別々に記述すること「も」可能であろう。しかしながらここで強調したいのは、そのような記述も恣意的な要素で、他の記述の形式も可能であるということである。言い換えれば、本研究ではそれらをあえて同一平面で表現した時に、どのような性質が顕現されるのかを研究したいのである。

^{*23} 「力学系」は厳密には無入力のシステムの数理モデルを指す数学用語である。また入力が想定される場合は「入力あり力学系」といった形で、その入力の存在が強調される。本研究で構成される数理モデルは両者の解釈

実現する数理モデルの構成法を提案する。そしてこれまでの複雑系やシステム論的アプローチを踏襲・発展させ、設計された高次元非線形力学系の解析を通して、自律的な学習機能の背後に通底する普遍的な制約条件の解明を目論む。そして得られた知見は、高度な自律性を有する汎用人工知能の実現のための足掛かりになることが期待されるだろう。

1.5 Innate Training と自律的な学習機能

先述の通り、本研究では自律的な学習機能を備えた高次元非線形力学系の設計論を提案する。特に本研究では自律的な学習機能を部分的に実現した数理モデルとして、Innate Training [47] と呼ばれる高次元カオス力学系^{*24}を制御、活用する機械学習手法に着目する。Innate Training は、端的に言えばある入力を与えられた時、再現的にある軌道を一定期間生成させるように高次元カオス力学系の内部結合を学習させる手法である。この Innate Training は、先に定義した自律的な学習機能を実現する上で重要な特性を 2 点有している。まず Innate Training では、カオス力学系が予め用意され活用されるのみならず、学習後も系のカオス性が維持される。すなわち学習の前後で恒常的にカオス的であり続け、入力を与えられなくても常に内部状態が変化しつづける。このようなカオス性は、システムの自発性を実現する駆動要因として役立てられることが期待される（要件 1）。^{*25}また Innate Training では、学習の際の教師軌道として自らが過去に出した複雑な高次元カオス軌道が採用される。系がカオス的であるため、この教師軌道はシステムの内部状態を反映した内的なものであり、その意味で学習対象を自ら構築しているといえる（要件 2）。加えてその自発的な活動の中で入力に対する定型的な活動パターンが埋め込まれるという意味で、Innate Training は脳の学習様式に抽象的に類似 [48] しており、自律的な学習システムのプロトタイプとして Innate Training は適している。

一方で Innate Training 全体を俯瞰した時、先に述べた自律的な学習機能の要件に抵触する要素がいくつか含まれている。本論文では以下の 4 点を指摘する。

改善点 1 入力時系列が外的な機構で生成・保持される（要件 3 に抵触）

改善点 2 入力が設計者によって受動的に与えられる（要件 1, 2 に抵触）

改善点 3 学習アルゴリズムが独立に計算される（要件 3 に抵触）

改善点 4 ハイパーパラメータが設計者によって決定される（要件 1, 2, 3 に抵触）

が可能である。すなわち環境を含む全体の中の部分系として学習システムをみなすならば前者として解釈できるし、環境との学習システムの相互作用を入出力と捉えるならば後者の表現が適当である。この点に留意しつつ煩雑さを避けるため、構成された数理モデルまたはニューラルネットワークを表現する単語として一貫して「力学系」や「非線形力学系」、あるいはもっと単純に「系」を使用する。また「高次元力学系」は特に次元数の大きいシステムを指す。

^{*24} 非線形力学系の中で、カオス性をもつ力学系のクラス。

^{*25} ここで注意したいのは、たとえ系がカオス的であっても、学習の停止と進行に関しては設計者によって制御される点である。すなわちその意味で、Innate Training は依然として自発的ではないことがここでは強調される。

まず Innate Training では入力として一種の過渡的入力を与えられるが、この時系列情報は系の外の機構で生成・保持される。そしてその生成・保持機構は学習システムと独立して存在しており充足的ではない（要件 3 に抵触）。*²⁶次に Innate Training では、入力が設計者によって受動的に与えられる。このような受動性は、学習システム全般に必然的に見られる特性であるが、ここではそのあり方や度合いが問題とされる。特に与えられる入力の種類の選択が設計者によって制御されており、それらは学習システムによって自己決定されていない（要件 2 に抵触）。またそのような受動的な入力の制御の仕方は、自発性を損ないかねない（要件 1 に抵触）。*²⁷加えて Innate Training では再帰最小二乗（Recursive Least Square, 以下 RLS）と呼ばれる学習アルゴリズムが採用されるが、それは外的な機構として学習システムとは独立に計算され、自己充足的でない（要件 3 に抵触）。そして Innate Training では、学習に関して設計者によって決められる数多くのハイパーパラメータが存在する。第一に教師軌道の長さ、すなわち学習の停止や進行が設計者によって制御されるが、そこでは恒常的に学習が進行しない（要件 1 に抵触）。第二に Innate Training では教師軌道として自ら生成した高次元カオス軌道が採用されているが、その教師軌道の初期値の選び方は設計者によって恣意的に選ばれている。通常その初期値はランダムに選択され特に制約はないとはいえ、その選定の過程において設計者による恣意性の混入を免れない（要件 2 に抵触）。そして第三に入力時系列同様に、これらのハイパーパラメータの設定や教師軌道そのものが外部の記憶装置に保持される（要件 3 に抵触）。

本研究ではこの 4 点の改善点を解消し、高次元非線形力学系の状態変化、すなわちダイナミクスの上で Innate Training が実現された自律的な学習システムの構成を目指す。特にニューラルネットワーク*²⁸を用いて高次元非線形力学系を構成し、その解析を通じて自律的な学習機能に要求される力学系的制約条件を考察する。

1.6 研究の目的

本研究では Innate Training と呼ばれる機械学習手法に着目しその拡張を行うことで、自律的な学習機能を備えた高次元非線形力学系の設計論を構築する。そしてその設計論を基に、実際に自律的な学習機能を有するニューラルネットワークを構成し、その自律的な学習システムとしての妥当性を示す。また構成された系の特性から、自律的な学習機能の背後に通底する普遍的な制約条件の解明を目論む。

*²⁶ 特に計算機では、CPU 上の演算で時系列情報が生成され、メモリに保持された上で再利用される。

*²⁷ 無論環境全てを部分系としてみなすことで、環境をも内包する学習システムを考えることができるが、そこでは完全に学習の対象が内在化され意味付けが困難になる。したがって、ここで企図されるのは、環境からの受動性の度合いを無にするのではなく、軽減する方法である。

*²⁸ 機械学習で用いられる数理モデルで、再帰的に結合されることで高次元非線形力学系的一种とみなされる。特に本研究では活性化関数として \tanh が用いられるものを使用する。

1.7 研究の意義

まず従来の観点とは異なり，自律的な学習機能が高次元非線形力学系のダイナミクスとして取り扱われる．言い換えれば，構成される高次元非線形力学系は，自律的な学習機能が封入された数理モデルとみなすことができる．そしてその力学系的特性は，自律的な学習機能を成立する上で重要な役割を果たしていると考えられ，その系統的な解析を通して明らかにされることが期待される．逆に，このような自律的な学習機能に対する理解の仕方は，数理モデルの実際の構成によってはじめて可能になる．その意味で本研究は生物知能，特にその学習機能に関して，新たな数理的アプローチを開拓しているといえよう．

同時に本研究は一貫して抽象的なレベル^{*29}で構成が行われ特定の系に限定されない．言い換えれば，得られる知見は普遍的であることが期待される．この普遍性ゆえに，例えば自然現象のダイナミクスの中に自律的な学習機能を抽出する，物理系の上で自律的な学習システムを実装するといった，特定の系に限定されない幅広い展開と運用が可能となる．また汎用人工知能の実現には実環境中で学習するシステム，すなわち実ロボットの実装が期待されるが，そのような実ロボットの設計の文脈においても，同様の理由から本研究の知見が効果的に威力を發揮すると考えられる．

加えて本研究では自律的な学習機能の自己充足的な構成が試みられる．特に非線形関数 (\tanh) を活性化関数とするニューラルネットワークが構成されるが，これはいわば \tanh が「身体性」として与えられた状況と解釈できる．^{*30}このような身体性は制約として働く反面その系，すなわち身体性に由来する予期せぬ効果をもたらす．そしてその効果はある身体性の下で構成して初めて得られる「副産物」で，その活用は構成論的アプローチの別の醍醐味である．^{*31}

1.8 論文の構成

まず本章では本研究の背景，動機ならびに目的を説明した．特に自律的な学習システムが満たすべき 3 点の要件を，機械学習手法と生物知能との比較を通して議論した．本研究では先述

^{*29} 力学系理論のレベル．

^{*30} すなわち生物とその学習機能とその物理法則に制約されるように，構成される自律的な学習システムは \tanh に制約されている．

^{*31} 例えば Thompson[49] は FPGA (Field Programmable Gate Array, 設計者が構成を自在にプログラムできる集積回路) を進化アルゴリズムによって最適化し，1 kHz と 10 kHz の音を識別する回路の設計を試みた．その結果入力部と出力部が物理的に断線した電子回路が得られたが，得られた回路はそれにもかかわらず正しく機能した．Thompson はこの結果について，電子回路が配線の代わりに回路間の電磁結合を活用していたと解釈したが，この状況では電磁結合という FPGA 特有の特性が引き出され活用されている．そしてその「副産物」は明らかに設計者によって予見されていない想定外の特性である．本研究では特に， \tanh という「身体性」で実装された時に副次的に表れる特性の活用が期待される．

の通り **Innate Training** を拡張し、自律的な学習機能が一貫して実現された高次元非線形力学系の設計論を構築する。次章以降は以下のように構成される。まず第 2 章では **Innate Training** の手法および特色に関して説明する。また **Reservoir Computing (RC)** や **Echo State Network (ESN)** [50], **RLS** アルゴリズム等、以降共通して現れる設定の説明ならびに定式化を行う。加えて先に導入された 3 要件に関して、**Innate Training** に関する 4 点の改善点を策定する。第 3 章, 第 4 章, 第 5 章, および第 6 章はその 4 点に対応する。まず第 3 章では **Input Reservoir** という概念を導入し、**Innate Training** における過渡的入力が高次元非線形力学系の部分系内の緩和過程で表現できることを示す。第 4 章では **Innate Training** では外的に制御されている入力の切り替えを、**Input Reservoir** ならびに高次元カオス力学系が生成するカオス軌道を活用し、線形閉ループを介して高い操作性で設計できる手法を提案する。第 5 章では、**All-In-One Reservoir (AIOR)** と呼ばれる概念および **Neuralizing** と名付けられる手法を提案する。そして **Innate Training** で用いられる学習アルゴリズムそのものが埋め込まれたニューラルネットワークである **Neuralized All-In-One Reservoir (NAIOR)** の設計手法を提案し、その有効性と妥当性を議論する。第 6 章では、**Innate Training** における教師軌道の生成に関する外的な要素が、学習システムにおいて内在化されるように拡張された **Developmental Innate Training (DIT)** を提案する。そして **Innate Training** 同等の学習の効果が **DIT** でも得られることを確認した上で、**DIT** の動的特性を解析する。最後に第 7 章では、ここまで導入された **Input Reservoir**, **Neuralizing**, **AIOR**, **DIT** を統合し、自律的な学習機能が埋め込まれた高次元非線形力学系である「統合学習システム」を実際に構成する。そして設計された統合学習システムの、自律的な学習システムとしての妥当性が検証される。第 8 章では、これらの一連の研究で構築された設計論を総括し、得られた知見とその意義を論じる。そして構築された高次元非線形力学系に関して、その解析および応用の具体的な指針を今後の研究の展望として議論する。

第 2 章

Innate Training—その手法と特色

本研究では第 1 章で述べられた通り，高次元カオス力学系を調整・制御する機械学習手法である Innate Training を基盤とし，その拡張により自律的な学習機能を有する高次元非線形力学系の設計論の構築を目指す。

本章ではまず必要な前提知識である RC と ESN を説明し，以降本論文で共通して登場する数式を導入する。次に，代表的な学習アルゴリズムとして Ridge 回帰と RLS アルゴリズム，ならびに RLS アルゴリズムを活用した FORCE (First-Order Reduced and Controlled Error) 学習とその動作原理を説明する。その上で Innate Training の手法について説明し，RC における他手法との比較を通して，その特色を複数の観点から議論する。これらを踏まえ，第 1 章で議論した要件に基づき，自律的な学習システムとして Innate Training が不十分な点を指摘する。最後にこれらの指摘を基に，自律的な学習システムの構成に向けた研究指針を提示し，次章以降の展開を説明する。

2.1 Reservoir Computing とその思想

RC は Recurrent Neural Network (RNN) を用いた機械学習手法である。RNN は内部に再帰的結合を有するニューラルネットワークの一種であり，非線形関数を活性化関数として採用するため，高次元非線形力学系とみなせる。そして過去の入力の情報を内部状態として保持・表現できるため，一般に時系列データの予測・生成に活用される [51]。

RNN の学習は通常勾配降下法による内部結合パラメータの調整によって行われる。再帰的な構造を有するため目的関数に対する勾配の計算は，BackPropagation Through Time (BPTT) 法 [52] や Real-Time Recurrent Learning 法 [53] 等を用いて，時間方向に展開して達成される。この勾配降下法による学習では多くの場合，RNN 内の全パラメータを調整する方式が採用される。RC は，これらの勾配降下法による手法とは異なり，RNN 内部の相互作用パラメータを

調整しない。代わりに Readout と呼ばれる外付けのモデル^{*1}を調整し、RNN の内部状態から所望の軌道を出力させる（図 2.1A, B 間の比較を参照せよ）。換言すれば、入力応答すなわち RNN 内で情報処理され変換された軌道をそのまま^{*2}活用する操作に等しい。このように RC は相互作用パラメータを調整しないことで、RNN のダイナミクスを抑圧せずにむしろ積極的に活用する学習様式であるといえる。そしてこの際活用される RNN、すなわち高次元非線形力学系は Reservoir と呼ばれる。この Readout のみの単純な学習手法にもかかわらず、周期的非カオス的なリミットサイクル（Limit Cycle）に加え、非整数次元の不変集合をもつストレンジアトラクタ（Strange Attractor）^{*3}の埋め込みが可能であり [54, 55]、高次元力学系を少ないパラメータの調整で制御できる手法として RC は近年脚光を浴びている [51]。

2.2 Echo State Network とその定式化

RC では、RNN のモデルとして ESN [50] が頻繁に用いられる。ESN は線形変換と非線形関数の単純な組み合わせで表現できる数理モデルで、RNN の中でもっとも単純な形式に分類される。以下典型的なケースとして、離散時間 $T = \{\dots - 2, -1, 0, 1, 2, \dots\}$ とその上で定義される ESN を考える。この時 ESN の時間発展方程式は以下の式で記述される。

$$\mathbf{x}(t+1) = \mathbf{H}(g\mathbf{J}\mathbf{x}(t) + \mathbf{u}(t)) \quad (2.1)$$

ここで $\mathbf{x} : T \rightarrow \mathbb{R}^N$ は ESN の N 次元の状態、 $\mathbf{H} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ は活性化関数、 $g \in \mathbb{R}$ は非線形パラメータ、 $\mathbf{J} \in \mathbb{R}^{N \times N}$ は内部結合に対応する線形行列、そして $\mathbf{u} : T \rightarrow \mathbb{R}^N$ は力学系に加わる入力を意味する。線形行列 \mathbf{J} はしばしば正規分布 $\mathcal{N}(0, \frac{1}{N})$ よりサンプルされ、スペクトル半径が 1 ($\rho(\mathbf{J}) = 1$) になるように正規化される [56]。他の機械学習での文脈同様、RC でも関数近似に関しての万能性、すなわち Universal Approximation Property (UAP) が肝要となる。この万能性を保証するため RC では系の Echo State Property (ESP) [50, 57, 58] が前提条件となる。^{*4}特に ESN では UAP を満足させるため、活性化関数 \mathbf{H} として \tanh ^{*5}が頻繁に使用される。また ESP を満たすため、特に $g < 1.0$ に設定され、系の非カオス性が保証された ESN (Non-Chaotic ESN) が使われる [59]。

また式 (2.1) の離散的な ESN の代わりに、以下の $T = \mathbb{R}$ 上で定義される連続的な ESN を採

^{*1} 通常線形パラメータが用いられる。

^{*2} 非線形変換が加えられないという意味。

^{*3} 例えば Lorenz 系や Mackey-Glass 系によって生成されるアトラクタがこれに該当する。

^{*4} 大まかに述べれば、ESP はある入力での系の漸近状態収束の条件である。UAP や ESN の定義および RC における重要性に関する議論は付録 A において補足される。

^{*5} 以下特に注記がなければ、Element-Wise なもの、すなわちベクトルの各要素がそれぞれ変換される写像が意味される。

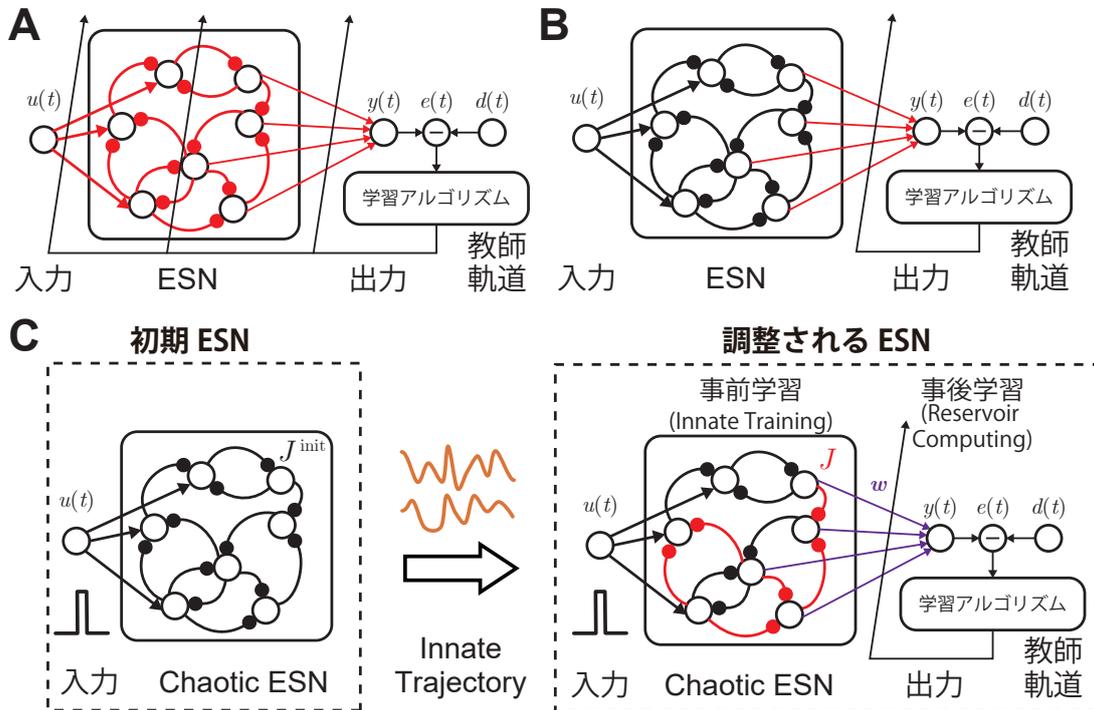


Fig. 2.1 RC および Innate Training の模式図. (A) 通常の勾配降下法に基づく RNN の学習様式. 赤色の線は調整 (学習) される相互作用の箇所を赤色で示している. 全ての相互作用が, 教師データ $d(t)$ と出力 $y(t)$ の誤差 $e(t)$ を最小化されるように学習される. 目的関数に関する勾配は学習アルゴリズム, 特に BPTT 等の勾配降下法によって計算される. (B) RC の学習様式. 勾配降下法による方法とは異なり, RNN 内の相互作用パラメータは固定化される. 代わりに内部状態を変換する開ループ系である Readout (通常線形) 内のパラメータのみを調整する. またこの固定化された RNN は Reservoir と呼ばれる. (C) Innate Training の模式図. Innate Training では, 高次元カオス力学系を用意し, 自らが過去に出したカオス軌道 (Innate Trajectory) を現的に生成されるように内部結合 J を学習する (事前学習). その後得られた ESN を Reservoir として活用し所望の出力を生成させる (事後学習).

用する場合がある.

$$\tau \dot{\mathbf{x}}(t) = -\mathbf{x}(t) + \mathbf{H}(gJ\mathbf{x}(t) + \mathbf{u}(t)) \quad (2.2)$$

この系は Leaky Echo State Network (Leaky ESN) と呼ばれ, より時定数の長い時系列を学習・生成する際に頻繁に用いられる. 特に数値シミュレーション上では, Euler 法によって離散系に直された以下の形式で計算される.

$$\mathbf{x}(t+1) = \left(1 - \frac{\Delta t}{\tau}\right) \mathbf{x}(t) + \frac{\Delta t}{\tau} \mathbf{H}(gJ\mathbf{x}(t) + \mathbf{u}(t)) \quad (2.3)$$

この離散系に変換された Leaky ESN 系の性質は [60] において詳細に調査されている.

2.3 Readout の学習アルゴリズム

先述の通り RC では、Readout と呼ばれる外付けのモデルの学習により、Reservoir の内部状態 $\mathbf{x}(t)$ を変換して所望の出力を生成させる。すなわち出力 $y(t)$ は、Readout を変換写像 $\phi: \mathbb{R}^N \rightarrow \mathbb{R}$ とおいた場合、以下の式で表される。

$$y(t) = \phi(\mathbf{x}(t)) \quad (2.4)$$

特に Readout が線形の場合、 ϕ の代わりに線形結合パラメータを $\mathbf{w} \in \mathbb{R}^N$ と設定し、以下の式で表現できる。

$$y(t) = \mathbf{w}^T \mathbf{x}(t) \quad (2.5)$$

RC ではこの線形な Readout が主に用いられる。以下 Readout を線形なものに限定し議論を進める。RC を含む、連続出力を扱う教師あり学習の枠組みでは、出力 $y(t)$ と教師軌道 $d(t)$ の誤差を最小化する問題が扱われる。換言すれば、以下の式で定義される L^p ノルムの目的関数 $C^p(\mathbf{w})$ を最小化するような \mathbf{w} を求める問題に帰着される。

$$C^p(\mathbf{w}) = \langle \|e(t)\|_p \rangle \quad (2.6)$$

$$= \begin{cases} \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T \|z(t) - d(t)\|_p dt & (\text{連続系}) \\ \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^T \|z(t) - d(t)\|_p & (\text{離散系}) \end{cases} \quad (2.7)$$

RC では特に L^2 ノルムを採用した $C^2(\mathbf{w})$ が用いられる。本節ではこの $C^2(\mathbf{w})$ の最適化の際に RC において用いられる Ridge 回帰、RLS アルゴリズム、および RLS アルゴリズムを活用した FORCE 学習を説明する。

2.3.1 Ridge 回帰

Ridge 回帰は Offline 学習の一種で、蓄積されたデータを基に線形結合パラメータ \mathbf{w} が計算される。具体的には L ステップにわたる内部状態のデータ行列 $X \in \mathbb{R}^{L \times N}$ と対応する教師軌道 $Y \in \mathbb{R}^L$ が手元にある時、推定パラメータ $\hat{\mathbf{w}}$ は以下の式で計算される。

$$\hat{\mathbf{w}} = (X^T X + \alpha I)^{-1} X^T Y \quad (2.8)$$

ここで $\alpha \in \mathbb{R}$ は正則項と呼ばれる定数で、過学習を防ぐ目的で導入される。またこの操作は以下の目的関数 $C_\alpha(\mathbf{w})$ を最小化するパラメータ $\hat{\mathbf{w}} = \operatorname{argmin} C_\alpha(\mathbf{w})$ を求める操作と等価である。

$$C_\alpha(\mathbf{w}) := C^2(\mathbf{w}) + \alpha \|\mathbf{w}\|^2 \quad (2.9)$$

換言すれば、この正則化項の導入で過学習により \mathbf{w} のノルムが大きくなりすぎるのが防がれる。また勾配降下法等の反復的な計算を要するアルゴリズムとは異なり、Ridge 回帰では最適なパラメータが陽に求まるため、One-Shot^{*6}で学習が完了する。この特性ゆえに、RC では Ridge 回帰が広く使われる。

2.3.2 RLS アルゴリズム

一方で RLS アルゴリズムは、データの追加に対して逐一 \mathbf{w} を更新するアルゴリズムで Online 学習に分類される。RLS アルゴリズムでは以下の目的関数を最小化する $\mathbf{w}(t)$ が求められる。

$$C_\mu(\mathbf{w}(t)) := \sum_{i=0}^{t-1} \mu^{t-1-i} e^2(i) \quad (2.10)$$

ここで $\mu \in \mathbb{R}$ は忘却項と呼ばれる項で $0 \leq \mu \leq 1$ を満たすように設定される。特に $\mu = 1$ の時に C_μ^2 と等価になる。^{*7}Ridge 回帰同様、 L ステップにわたるデータを蓄積し逆行列を計算することで最適な $\mathbf{w}(L)$ を求められるが、この操作は各時刻において毎回 $O(N^3)$ の計算量が生じる。そこで RLS アルゴリズムでは逆行列補題 [61] を駆使して、 $O(N^2)$ の計算量で逆行列の更新を完了させる。具体的には、補助変数として $P(t) \in \mathbb{R}^{N \times N}$ を導入し以下の漸化式でパラメータを更新する。

$$P(0) = \frac{I}{\alpha} \quad (2.11)$$

$$P(t+1) = \mu^{-1} P(t) - \frac{P(t)\mathbf{x}(t)\mathbf{x}^T(t)P^T(t)}{\mu + \mathbf{x}^T(t)P(t)\mathbf{x}(t)} \quad (2.12)$$

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \mu^{-1} (d(t) - y(t)) P(t)\mathbf{x}(t) \quad (2.13)$$

ここで $P(t)$ の初期値において、正則化項 α が設定されることに注意されたい。なお漸化式中の $P(t)$ は陽に計算でき、以下の式で表される。

$$P(t) = \left(\sum_{i=0}^{t-1} \mu^{t-1-i} \mathbf{x}(i)\mathbf{x}^T(i) + \mu^t \alpha I \right)^{-1} \quad (2.14)$$

また Ridge 回帰等の Offline 学習とは異なり、RLS アルゴリズムでは各時刻において設計者によって学習の進行が制御される。特に実用上では、何らかの基準で途中で学習が停止されなければならないが、一般的には誤差 $e(t)$ の大きさや $\mathbf{w}(t)$ の更新量を基に学習が停止される。

^{*6} 一回の学習データの提示で、という意味。

^{*7} また通常ほとんど考慮されないが、 $\mu = 0$ の時は最小二乗法と等価とみなせる。

2.3.3 FORCE 学習

先述の通り RC の条件として、Reservoir として活用される力学系の ESP が保証される必要がある。これは Reservoir, すなわち入出力関係を学習する関数近似器として、カオス力学系は本来適さないことを意味する。一方でカオス力学系のダイナミクスを制御し、活用する学習アルゴリズムとして FORCE 学習 [55] が提案された。特に FORCE 学習はカオス的な ESN (Chaotic ESN) に付与された線形な閉ループを学習する手法である。具体的には以下の式で表されるような Leaky ESN ($g = 1.5$) を用い、線形閉ループのパラメータ \mathbf{w} を学習させる。

$$\tau \dot{\mathbf{x}}(t) := -\mathbf{x}(t) + \mathbf{H} \left(g \mathbf{J} \mathbf{x}(t) + \mathbf{U}_{\text{feed}} \mathbf{w}^T \mathbf{x}(t) \right) \quad (2.15)$$

特に FORCE 学習ではこの \mathbf{w} の学習において、RLS アルゴリズム ($\mu = 1$) が用いられる。^{*8} それにもかかわらず、ストレンジアトラクタまで含めた様々なアトラクタが設計できるほど高い操作性を有している [55]。この FORCE 学習の高い操作性は、閉ループ系の入力により Noise-Induced Order [63] と同等の分岐が発生したことによる ESP の回復に起因すると推測される。実際 Sussillo ら [55] による実験では、学習の開始にともない Chaotic ESN の非カオス状態への遷移が観察される。加えて興味深いことに、FORCE 学習では Non-Chaotic ESN に比して Chaotic ESN の方が、高い精度でアトラクタが設計できることが報告されている。一方で、このカオス性の FORCE 学習の精度に対する寄与原理は十分に明らかにされておらず、今後の研究が期待される。まとめると FORCE 学習は、カオス力学系上の閉ループを RLS アルゴリズムによって調整し、所望の軌道を設計する手法として特徴づけられる。

2.4 Innate Training とその特色

ここまで RC および RC における代表的な学習アルゴリズムを説明した。この節では本研究の基幹に位置付けられる Innate Training [47] の内容とその特色を説明する。Innate Training は FORCE 学習をさらに拡張した発展的な学習方式である。大まかに述べれば、ある入力を与えられた時、自らが過去に出力した高次元カオス軌道を再現的に生成させるように Chaotic ESN の内部結合を学習し、得られた Chaotic ESN とその高次元カオス軌道を Reservoir として活用する手法である。特に前者の内部結合の学習は事前学習、後者の過程は事後学習と呼ばれる。

まず内部結合の学習、すなわち事前学習の方法に関して説明する。ここでは具体例として

^{*8} なお $\alpha = 1.0$ が正則化項として主に使用されるが、他の値も採用されうる。実際 [55] では $\alpha = 1.0$ の他、2.0, 40.0, 80.0 といった値が実験において使用されている。また本来 FORCE 学習はその名前の通り目的関数の形式で定義づけられ、その学習アルゴリズムは RLS アルゴリズムに限定されずに他のものも採用されうる。しかし [55] で検証されているように、カオス力学系の閉ループ系の学習において、他学習アルゴリズムは RLS アルゴリズムほどの操作性はなかった。この点に関して、アルゴリズムの改良が [62] で試みられている。

Innate Training において用いられるスパースな Leaky ESN (内部結合 J が零成分を含む)^{*9}を考える. この ESN において i ($1 \leq i \leq N$) 番目のノードの時間発展方程式は以下の式で表現される.

$$\tau \dot{x}_i(t) = -x_i(t) + H_i \left(\sum_{j=1}^N g J_{ij} x_j(t) + u_i(t) \right) \quad (2.16)$$

ここで i 番目のノードに投射するノードの集合を $B_i := \{j \mid J_{ij} \neq 0\}$ とおけば, $x_i(t)$ は $x_j(t)$ ($j \in B(i)$) の線形和が投射されていると解釈できる. 換言すれば J_{ij} ($j \in B(i)$) は i 番目のノードに対する一種の線形な Readout で, $x_i(t)$ はこの Readout の出力であるとみなせる. したがって, RC の手法を用いて, この J_{ij} ($j \in B(i)$) を同様に調整し, Chaotic ESN の内部結合を学習できる.

さてこの時間問題になるのが学習に用いられる教師軌道の設計方法である. 実は内部結合が, 閉ループ系の Readout と等価にみなせるという発想はすでに FORCE 学習の論文において示されており, ここでは教師軌道の候補がいくつか提示されていた [55]. しかしながら全ての結合を学習させる方法は基本的には不安定であり, その設定には何かしらの工夫が必要であることが指摘されている [64]. Innate Training の独創的な点は, 学習前のネットワークがある入力 $u(t)$ の下, 一度出力した高次元カオス軌道を教師軌道として設定する着想にある. 特に Innate Training の事前学習では, 学習前のネットワークが生成する高次元カオス軌道 $\mathbf{x}^{\text{target}}(t)$ が教師軌道^{*10}として保存され, 以下の式で表される目的関数が最適化される.

$$C_J = \left\langle \frac{1}{L} \int_0^L \|\mathbf{x}(t) - \mathbf{x}^{\text{target}}(t)\|^2 dt \right\rangle \quad (2.17)$$

この時 L は教師軌道の長さ, ブラケットはサンプル平均を表す. この目的関数の下 Innate Training の事前学習では, 複数の Readout の集合であると解釈された内部結合に対して, それぞれ FORCE 学習が並列に実行される (図 2.1C). またもし学習が正しく達成された場合, Chaotic ESN はある入力関数が投射された時に, 教師軌道 $\mathbf{x}^{\text{target}}(t)$ を再現的に生成されることが期待される. さらに内部に FORCE 学習を用いているため, この事前学習は Online 学習の一種である. 特に Innate Training では, 時刻 $t = 0$ に初期値 $\mathbf{x}(0)$ がランダムにサンプリングされ $t \in [0, L]$ の期間, 内部結合が調整される操作を 1 エポック (Epoch) とし, それが複数回数繰り返されることにより事前学習が達成される. そしてこの事前学習は, ある指定されたエポック数が完了されるか, 内部結合 J の更新量や誤差 C_J が所定の閾値以下になるまで行われる.

図 2.2 は事前学習前と事前学習後の, 入力に対する系の応答を示したものである. 初期値の異なる 10 軌道が重ね描かれることで, 軌道の再現性が可視化されている. 系そのものがカオ

^{*9} Innate Training [47] では結合密度 (非零成分の割合) 0.05 の 800 次元の Chaotic ESN が使用された.

^{*10} またその生成のさせ方から, この高次元カオス軌道は Innate Trajectory と呼ばれる.

的であるため、いずれも自発的に複雑な軌道が生成されているが、事前学習後のネットワーク（下図）では単一矩形波の入力後、一定の期間教師軌道（点線）に沿うように軌道が生成されている様子が観察される。またカオス力学系を活用する別の手法である FORCE 学習においては、閉ループ系によっていわばカオス性を抑圧する形でダイナミクスが埋め込まれるが、Innate Training では事前学習後もカオス性が維持され続ける。実際図 2.2 でも学習後の ESN においても、そのカオス性のため一旦同期した軌道が拡散している様子が観察できる。

また事後学習ではこの再現的に生成される高次元カオス軌道を活用するために、外的な線形モデル^{*11}が学習される。一般に高次元非線形力学系は入力に対する高い分離能を有する。すなわち入力を高次元系に投射されることで、線形モデルでも非線形な入出力関係を学習させやすくなる [65]。また Innate Training では高次元カオス軌道が再現的に生成されるが、これは高次元カオス軌道の局所的に ESP が回復されていると解釈できる。したがって、この埋め込まれた高次元カオス軌道は、ESP と高い分離能を局所的に兼ね備えており、Reservoir として入出力関係の生成に役立てられることが期待される。実際 [47] では、Chaotic ESN によって再現的に生成される高次元カオス軌道を用いて、所定の矩形波入力の後、一定期間筆記体の軌跡を生成する二次元出力を、単純な線形モデルのみで実現される様子が示されている。この線形モデルは Ridge 回帰を用いて One-Shot 的に得られる。すなわち所望の出力軌道が用意されれば容易に計算される。

このように、Innate Training はある入力に対して一定期間再現的にカオス軌道を生成する高次元カオス力学系を構築する手法である。そして再現的に生成される高次元カオス軌道は、Reservoir として入出力関係の設計に活用される。

2.5 自律的な学習システムとしての Innate Training

ここまで Innate Training の学習様式とその特色について説明した。ここでは、第 1 章で議論された自律的な学習システムとしての Innate Training の適性を議論する。まず先述の通り Innate Training では、カオス力学系が活用されるのみならず、学習後も系のカオス性が維持される。すなわち学習の前後で Innate Training は恒常的にカオス的であり続ける。学習の停止と進行そのものは依然として設計者によって制御され自発的でないものの、Innate Training は自発的な構成を実現しやすい状況にあるといえる（要件 1）。また Innate Training では、自らが過去に出した複雑な高次元カオス軌道が学習の際の教師軌道として採用される。系がカオス的であるため、この教師軌道は完全にシステムの内部状態とその相互作用の様態を反映した内的なものであり、その意味で学習対象を自ら構築しているといえる（要件 2）。

加えて Innate Training は抽象的な意味で脳の学習様式の一側面を反映していると考えられ

*11 すなわち線形な Readout.

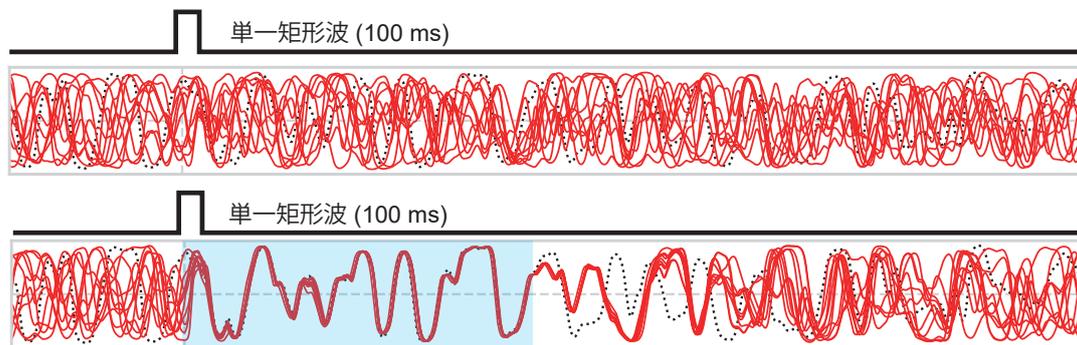


Fig. 2.2 Innate Training の学習前後の比較. 異なる初期値から時間発展した 10 個の ESN の高次元カオス軌道に関して, 単一矩形波 (100 ms) を付与した際の挙動を重ね描いている. 初期 ESN によって生成された高次元カオス (点線) が教師軌道 ($\mathbf{x}^{\text{target}}(t)$) として採用され, $[0, L]$ の区間 (水色) で学習される. Innate Training は, ある入力 $\mathbf{u}(t)$ (この場合単一矩形波) の下, 様々な初期値に対して教師軌道 $\mathbf{x}^{\text{target}}(t)$ を再現的に出力するように内部結合を学習させる.

る. 先述の通り脳は無数の神経細胞から構成され, 様々なスケールでカオス性を示す高次元カオス力学系であると解釈できる. 一方で脳の神経活動のレパートリは可能空間と比してずっと限定的であり, かつ入力に応じて励起されるいくつかの典型的な活動パターンで構成されることが報告されている [48]. そしてこの励起される活動のパターンは環境との相互作用の結果構築されたものであるとみなせる. このように, どちらも高次元カオス力学系のダイナミクスを調整し, 入力に対応した構造をその軌道に反映させる点で, 抽象的な意味で Innate Training は脳の学習様式は類似しているといえる. このような生物学的な類似性は, 自律的な学習システムの要件として直接は要求されていないものの, Innate Training の自律的な学習システムとしての適性を支持するものである.

2.6 Innate Training の改善点

一方で Innate Training を俯瞰した時, 第 1 章において定義された自律な学習システムの要件を満たさない要素がいくつか散見される. 本研究では特に以下の 4 点を指摘する.

改善点 1 入力時系列が外的な機構で生成・保持される (要件 3 に抵触)

改善点 2 入力 that 設計者によって受動的に与えられる (要件 1, 2 に抵触)

改善点 3 学習アルゴリズムが独立に計算される (要件 3 に抵触)

改善点 4 ハイパーパラメータが設計者によって決定される (要件 1, 2, 3 に抵触)

まず Innate Training の事前学習では、ある種の過渡的入力^{*12}が毎エポック与えられる。この過渡的入力の時系列は、現在は系の外で生成されかつ外側の記憶機構（コンピュータのメモリ等）で保持される。これらの機構は学習システム、特に Chaotic ESN と分離しており、要件 3 に抵触している。Chaotic ESN と一貫してこれらの機構が表現されることで、この点は解消されると期待される。

次に Innate Training では現状、入力が設計者によって受動的に与えられる。このような受動性は、学習システム全般に必然的に現れる特性で完全には排除できないものの、将来的にその度合いを調整・軽減できることが望ましい。例えば Innate Training では複数種類の入力が許容され、現状ではその入力の種類の選択および切り替えが設計者によって制御されている。これは一種の教示とも捉えられるが、そのような要素が学習システム自身によって設計者から独立して制御されれば、要件 1 の自発性ならびに要件 2 の学習対象の自己決定・構成能がより高められた構成が実現される。

また Innate Training では内部結合が FORCE 学習を用いて調整され、そこでは複数の RLS アルゴリズムが並列に動作しているとみなされる。この RLS アルゴリズムは、Chaotic ESN とは完全に独立して存在し駆動される。結果的に RLS アルゴリズムが内部結合を調整する階層的な構成がとられるが、これらの要素は自己充足的ではなく、要件 3 に抵触する。Chaotic ESN と RLS アルゴリズムが一貫した RNN で表現されることでこの点は解消されるだろう。

加えて Innate Training では自らが生成した高次元カオス軌道が教師軌道として採用されている。この時その教師軌道のハイパーパラメータ、すなわち長さや初期値の選択は依然として設計者の手に委ねられている。これらは学習対象の自己決定性を要求する要件 2、ならびに生成機構の内包を要求する要件 3 に抵触している。また先述の通り、Innate Training の学習の停止はある基準のもとに設計者によって制御される。したがって、恒常的に学習が行われず要件 1 に抵触している。教師軌道が学習システムの中で生成され続け、かつ学習が恒常的に進行する構成において、これらの点は解消されうる。

2.7 自律的な学習システムに向けた拡張の指針

本研究では Innate Training を基盤に前節で指摘された改善点を克服することで、自律的な学習システムの構成を目指す。以下、各改善点に対応する具体的な研究の指針が示される。またこれらはそれぞれ第 3 章、第 4 章、第 5 章、第 6 章、ならびに第 7 章に対応する。

^{*12} Innate Training [47] では単一矩形波が使用される。

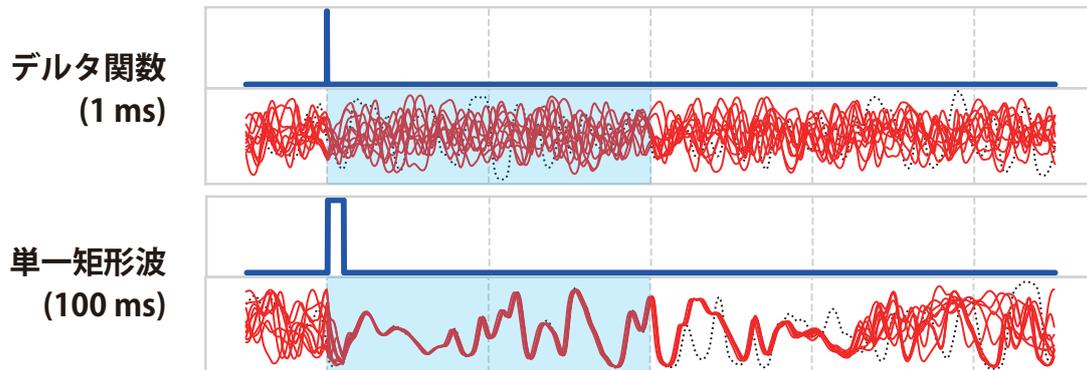


Fig. 2.3 Innate Training における過渡的入力的重要性. 図は Innate Training 後の Chaotic ESN の 10 軌道を重ね描いている. 上図では 1 ステップのデルタ関数, 下図では 100 ステップにわたる単一矩形波を使用し事前学習が行われた. 図から分かる通り学習がデルタ関数を採用した場合は, 再現的に高次元なカオス軌道が生成されず, その意味で学習に失敗している.

2.7.1 過渡的入力の内在化

まず Innate Training で導入される過渡的入力に関して, あらゆる入力時系列が許容されるわけではなく, 何らかの制約があると推測される. 例えば図 2.3 に示される通り, 時間的な構造を持たないデルタ関数を採用したケースでは学習は失敗する. このことは何かしらの入力の時間的構造が Innate Training において要請されることが示唆されるが, 入力の構造は固定化されており現時点でその仕様は不明である. そこで本研究では Innate Training において与えられる過渡的入力果たす役割を考察し, その制約を調査する. またその制約を満たす過渡的入力の生成機構を, ESN を用いて表現することで学習システムに内在化させる.

2.7.2 自発的な記号の切り替え

次に現在は設計者によって制御されている入力の種類の切り替えを, Chaotic ESN を用いて自発的に実現する方法を提案する. 特に Chaotic ESN のカオス性を活用し, 設計者から予測不能な形で, 複数の入力の候補の中から系自身が選択する自発的な構成を実現する.

2.7.3 学習過程の自己充足的表現

Innate Training では各内部結合を調整する機構として RLS アルゴリズムが採用される. こ

の RLS アルゴリズムは式 (2.12) から分かる通り、RNN とは全く別^{*13}の時間発展方程式で記述される。本研究ではその外的な機構として計算される RLS アルゴリズムを、同一の RNN 上のダイナミクスとして表現し、学習システムの自己充足性を向上させる。

2.7.4 教師軌道の自己決定・構成

Innate Training の教師軌道の初期値といったハイパーパラメータが、学習システムによって自己決定される形で Innate Training を拡張する。また学習の進行と停止の切り替えを無くし、恒常的に学習が進行する Innate Training の構成を提案する。そして拡張された Innate Training の動的特性を解析し、その機能性と応用を議論する。

2.7.5 手法の統合および解析

ここまでの手法を統合し、第1章で定義された要件を満たす形で、自律的な学習機能が実現された RNN、すなわち高次元非線形力学系を構成する。また得られた高次元非線形力学系の系統的な解析を通して、自律的な学習機能の背後に通底する普遍的な動的特性の解明を目標とする。

^{*13} \tanh の非線形性とは全く別の関数を使用されているという意味。

第 3 章

Input Reservoir の提案および Innate Training の学習性能への影響

本章では、Innate Training の学習に必要とされる入力の特徴が調査され、その上で入力時系列の生成機構を学習システムに内在化させる。特に Input Reservoir [66] と呼ばれる概念を導入し、Innate Training を成立させるような過渡的入力を Non-Chaotic ESN を用いてモデル化し、Input Reservoir としてある大きな ESN の部分系として内在化させる。さらに本章では、Input Reservoir として ESN の代わりにソフトボディを採用し、身体における柔軟な要素が与える脳の情報処理への寄与を考察する。

3.1 問題設定

3.1.1 要求される過渡的入力の特徴に関する考察

先述の通り、Innate Training はある入力に対するカオス力学系の応答を再現的にさせる手法である。この際重要な役割を果たすのが、ある強度で一定期間投射される過渡的入力の存在である。実際、先行研究では単一矩形波 [47] や音声スペクトルより生成された複数チャンネルの信号 [67] が Innate Training の入力として採用されている。また Innate Training ではあらゆる入力時系列が許容されているわけではない。例えば図 2.3 で示されるように、時間的な構造を全く持たないデルタ信号が学習に用いられても再現的な応答は獲得されなかった。加えて単一矩形波のように、入力強度が一定期間後に弱まるような過渡的な時系列の構造が学習において重要であると推測される。特に Chaotic ESN は入力強度の増加にともない、非カオス状態に遷移する分岐が生じる [68]。したがって、ある強度以上の入力が投射され続けた場合遷移により、Innate Training の特色であるカオス性が損なわれてしまう。これらをまとめると Innate Training では、ある一定の強度で投射されかつ一定期間後に消失するような過渡的な時系列の

構造が重要であると考えられる。

3.1.2 過渡的入力モデル化および Input Reservoir による内在化

ところでそのような過渡的入力、非カオス力学系の緩和過程によって表現できる。そこで本章ではまず過渡的入力の生成機構を Non-Chaotic ESN としてモデル化する。そして Input Reservoir と呼ばれる概念を導入し、Input Reservoir として Non-Chaotic ESN が内在化された ESN を構成する。

まず本章で導入される Input Reservoir を説明しよう。Input Reservoir は入力とカオス力学系の間を挟む力学系を指し、カオス力学系への一方向的な結合で特徴づけられる (図 3.1)。すなわち Input Reservoir が入力に対するフィルタとして時系列的な応答を生成し、その応答がカオス力学系にフィードフォワードに投射される。また Input Reservoir 自体もある大きな力学系における部分系と解釈することができる。すなわち逆に、ある力学系においてそのような一方向的に結合する部分系が見出された場合、その部分系を Input Reservoir とみなすことが可能である。

この Input Reservoir 自体は概念的なもので、カオス力学系への一方向的な結合が保たれている限りは、あらゆる力学系が当てはまる。^{*1}特に Input Reservoir とカオス力学系がある同一の機構で実現されるならば、それらが一貫して大きなカオス力学系の上で表現されているとみなされるだろう。そこで、Innate Training では Chaotic ESN がカオス力学系のモデルとして使用されるため、過渡的入力を表現するために Non-Chaotic ESN を Input Reservoir に採用しよう。すると、全体としては大きな ESN の部分系として、Input Reservoir およびカオス力学系が表現される。また Input Reservoir は非カオス的であるため、入力が与えられなければ緩和過程を生成したのち Input Reservoir のダイナミクスは停止し、カオス力学系への入力の投射が消失する。このように Input Reservoir として、過渡的入力の生成機構が一貫して表現された ESN を構成できる。

さて次節以降の実験の指針を述べる。本章ではまず、Non-Chaotic ESN としてモデル化された過渡的入力の性質と、Innate Training の学習性能の関係を調査する。そして Innate Training において要求される入力の性質の解明を目論む。具体的には Non-Chaotic ESN のスペクトル半径を調整し、生成される過渡的入力の長さや Innate Training の性能の関係を調査する。さらに発展的な話題として、生物の身体においてみられる柔軟な要素がもつ計算論的な機能性を議論するため、ソフトボディのセンサダイナミクス [69] を Input Reservoir として採用し実験する。生物の身体は一般に、皮膚や筋肉といった柔軟な要素を多く保有している。これまでソフトロボティクス分野において、物質のもつ柔らかさを活用して、しなやかかつ頑強な振る

^{*1} 特に Input Reservoir は必ずしも非カオス的である必要はないことに注意されたい。あくまで入力を反映した応答を生成する機構として機能している。

舞いを実現した実システムが数多く提案されている [70, 71]. また近年ソフトボディにおいて生み出される豊富な時系列パターンが計算資源として、ある種の記憶と非線形性を要する入出力関係の近似に活用できることが実証されている [72, 73, 74, 75, 76, 77].^{*2}本章では、これらの研究に立脚する形でソフトボディと脳・神経系との関係に踏み込みたい. すなわち脳・神経系が皮膚や筋肉等の柔軟な要素が覆われる構成は、生物において広く観察される普遍的なものである. そして、身体の柔軟な要素を外部の入力から過渡的入力に生成するフィルタすなわち **Input Reservoir** としてみなし、この生物において広く観察される構造がもつ役割を考察する.

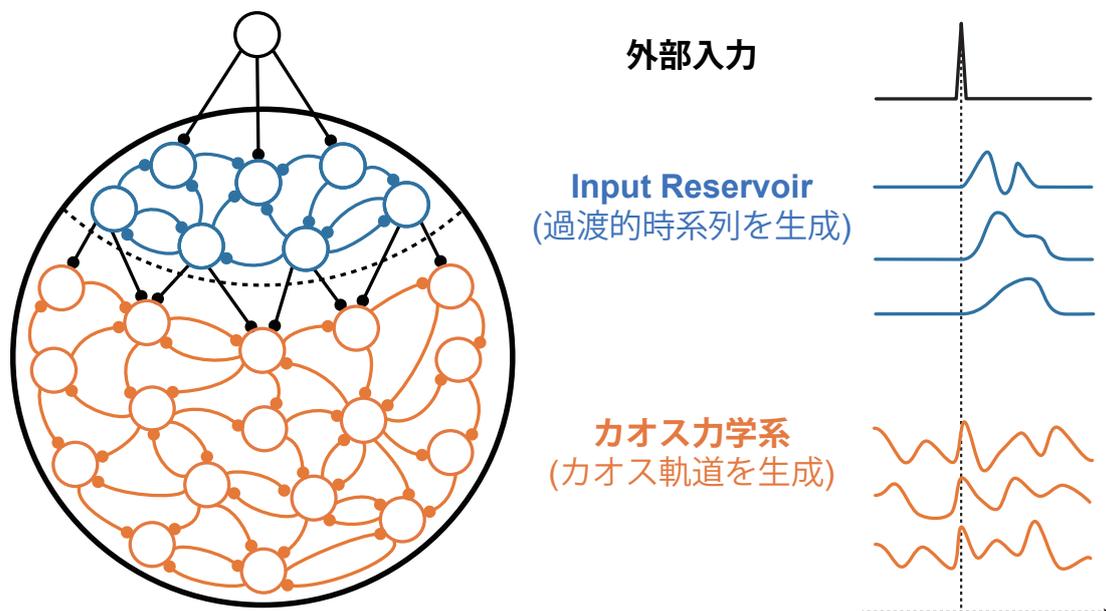


Fig. 3.1 Input Reservoir の概念図. Input Reservoir は外部入力とカオス力学系間のフィルタとして機能する. 特にこの Input Reservoir として非カオス力学系を採用した際, Innate Training で用いられるような入力に応じた緩和過程が生成され, カオス力学系に投射される. なおカオス力学系から Input Reservoir へのフィードバックループは存在しない. 今回はこの両者が ESN として表現される.

3.2 提案手法

構築される系の設定および評価手法に関して詳細に説明する. 本章では Input Reservoir として Non-Chaotic ESN を用意し, そこで発生した過渡的な時系列を Chaotic ESN に投射し, Innate Training によって Chaotic ESN の内部結合を学習する. すなわち, Input Reservoir で発生した緩和過程を, Innate Training において用いられる過渡的入力として活用し, その効果を検証する.

^{*2} すなわちここではソフトボディが Reservoir として活用されている.

3.2.1 Chaotic ESN および Innate Training の設定

まず Chaotic ESN は、以下の Leaky ESN の形式で記述される。

$$\tau \frac{d\mathbf{x}^{\text{ch}}}{dt}(t) = -\mathbf{x}^{\text{ch}}(t) + \tanh\left(g^{\text{ch}} J^{\text{ch}} \mathbf{x}^{\text{ch}}(t) + u_{\text{amp}} \mathbf{u}^{\text{ch}}(t)\right) \quad (3.1)$$

ここで $\mathbf{x}^{\text{ch}} \in \mathbb{R}^{N^{\text{ch}}}$ は Chaotic ESN の内部状態、 $J^{\text{ch}} \in \mathbb{R}^{N^{\text{ch}} \times N^{\text{ch}}}$ は内部結合、 $\mathbf{u}^{\text{ch}}(t)$ は Chaotic ESN に投射される入力^{*3}、そして $u_{\text{amp}} \in \mathbb{R}$ が投射強度を調整する係数である。 $N^{\text{ch}} = 800$ の Leaky ESN を用意し、カオス性を担保するため、実験では $\tau = 10.0$, $g^{\text{ch}} = 1.6$ の条件を採用した。内部結合行列 J^{ch} はスパースなランダム行列（結合密度 $p = 0.1$ ）で非零要素は $\mathcal{N}\left(0, \frac{1}{pN}\right)$ より生成される。^{*4}そして、以下の目的関数の下、この Chaotic ESN のノードの内半分を選択し、その前結合を Innate Training によって調整する。

$$C_{J^{\text{ch}}} = \left\langle \int_0^L \|\mathbf{x}^{\text{ch}}(t) - \mathbf{x}^{\text{target}}(t)\|^2 dt \right\rangle \quad (3.2)$$

ここで $\mathbf{x}^{\text{target}}(t)$ は教師軌道、 $L \in \mathbb{R}$ は教師軌道の長さ、ブラケットはサンプル平均を表す。本実験ではある初期値から $\mathbf{x}^{\text{target}}(t)$ をサンプルし、 $L = 2000 \text{ ms}$ ^{*5}の間、再現的に生成させるように学習させた。学習は 50 エポック行われた。

3.2.2 Timer Task および Timer Task Capacity

本章では Timer Task と呼ばれるタスクを用意し、Innate Training 後の系の時間的情報処理能力を評価する。Timer Task は系の内部状態 $\mathbf{x}(t)$ の線形変換のみで、 t_{peak} [ms] 後に遅れてピークが来るパルス状の波を再構築させるタスクである（図 3.2）。言い換えれば、内部状態の $\mathbf{x}(t)$ においてどれほど長く入力の情報が反映されているか Timer Task では評価される。実験では Readout \mathbf{w}_{out} を学習させ、以下の式で表されるパルス状の波を出力させる。

$$\mathbf{w}_{\text{out}}^T \mathbf{x}(t) \approx \exp\left(\frac{(t - t_{\text{peak}})^2}{2 \times 10^2}\right). \quad (3.3)$$

この学習は Ridge 回帰 ($\alpha = 1$) によって達成される。またその性能は、目標軌道と出力の間の R^2 値の 10 回平均により評価される。^{*6}これを Timer Task Function (TTF) $R^2(t_{\text{peak}})$ とし、それを積分することで Timer Task Capacity ($C_{\text{TT}} := \int_0^{\infty} R^2(t) dt$, 以下 TTC) を得る。なお数値計算上では TTC は $R^2(t)$ が十分 0 に近づく範囲 ($0 \leq t_{\text{peak}} < 10000$) で積分された。

^{*3} 後述の通りこの入力は Input Reservoir において生成される。

^{*4} これらの条件は [47] で使用されたものである。

^{*5} なお以降 Leaky ESN が登場する場合は、特記事項がない場合は式 (2.3) の形式で $\Delta t = 1$ の条件で計算される。また実環境システムとの対応を考え、その時間幅が 1 ms に設定される。

^{*6} 変数 x, y に関して $R^2(x, y) := \frac{\text{Cor}(x, y)^2}{\sigma_x^2 \sigma_y^2}$ と定義される。

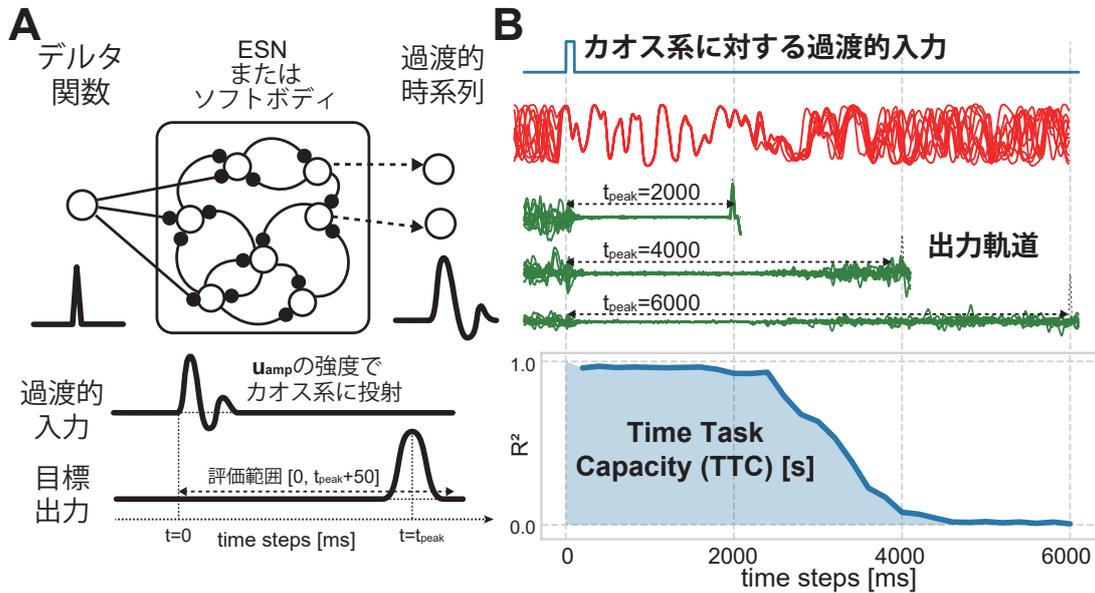


Fig. 3.2 実験系および Timer Task の説明. (A) 実験系の概要図. Input Reservoir で生成された過渡的入力のカオス力学系に投射し Innate Training を行う. この際本実験では Input Reservoir として (i) スペクトル半径 g の ESN および (ii) モータコマンド m で制御されるソフトセンサを使用した. 本実験では Timer Task を用いて系の時間的な情報処理能力を計測した. 具体的には入力を $t=0$ で与え ΔT 後にピークを持つ波を出力させることで, 入力情報の保持能力を計測する. (B) 実際の評価例および, TTC (Timer Task Capacity) の定義. 各 ΔT に対して Readout を更新し R^2 値を計測, その積算値として TTC が定義される.

3.3 結果

3.3.1 数値計算による評価

まず Non-Chaotic ESN を用いた数値実験の結果を述べる. Input Reservoir として以下の式で定義される Non-Chaotic ESN を用意した (図 3.2A).

$$\tau \frac{d\mathbf{x}^{\text{in}}}{dt}(t) = -\mathbf{x}^{\text{in}}(t) + \tanh\left(g^{\text{in}} J^{\text{in}} \mathbf{x}^{\text{in}}(t) + U^{\text{in}} \delta(t)\right) \quad (3.4)$$

ここで $U^{\text{in}} \in \mathbb{R}^{N^{\text{in}}}$ は一様乱数 $U([-1, 1])$ よりサンプルされるランダム行列である. $N^{\text{in}} = 200$ ノードの ESN を用意し, 非カオス性を担保するため, 実験では $\tau = 10.0$, $g^{\text{in}} < 1.0$ の条件を採用する. $\delta(t)$ はデルタ関数を再現しており時刻 $t=0$ の時のみ 1 を出力し, それ以外の時は 0 が出力される. Non-Chaotic ESN から Chaotic ESN へ投射される入力 $\mathbf{u}^{\text{ch}}(t)$ は以下の式で表される.

$$\mathbf{u}^{\text{ch}}(t) = U^{\text{ic}} \mathbf{x}^{\text{in}}(t) \quad (3.5)$$

ただし $U^{ic} \in \mathbb{R}^{N^{ch} \times N^{in}}$ は一様乱数 $\mathcal{U}([-1, 1])$ よりサンプルされるランダム行列である。本実験では投射強度 u_{amp} を変化させ、Innate Training の性能への寄与を TTC によって評価した。

まず、Innate Training におけるカオス性の重要性を評価するため、投射先の ESN のカオス性を変化させ評価した。具体的には $g^{ch} = 0.9, 1.6$ の二つのケースに関して TTC を計測した。なおこの時 Input Reservoir のパラメータは変化させない。図 3.3 ではその結果が示される。まず図 3.3A では共通の入力が与えられた時の Chaotic ESN ($g^{ch} = 1.6$) と Non-Chaotic ESN ($g^{ch} = 0.9$) のダイナミクスの様子が表示されている。図から明らかなように、Innate Training によって高次元の複雑な軌道が Chaotic ESN では入力に応じ再現されているのに対し、Non-Chaotic ESN では入力の影響が消失しある固定点に収束する様子が確認された。また Timer Task の学習においても、Chaotic ESN では正しくパルスが出力されているのに対し、Non-Chaotic ESN では上手く生成されなかった。結果的に図 3.3B より分かる通り、ある一定以上の投射強度 u_{amp} の範囲において、TTC において大きな差が開いた。この実験結果からまず、系のカオス性の Innate Training における重要性が示唆される。

次に、Input Reservoir 自体のパラメータを変化させ、その影響を調査した。まず Input Reservoir 自体の効果を確認するため、デルタ関数 $\delta(t)$ を直接 Chaotic ESN に投射する条件を用意した。この際 $\mathbf{u}(t)$ を構成するため、適当なランダム行列 U^{direct} を一様乱数 $\mathcal{U}([-1, 1])$ からサンプルして用意し $\mathbf{u}(t) = U^{direct}\delta(t)$ とおいた。また緩和過程の長さが与える学習性能の影響を調査するため $g^{in} = 0.1, 0.99$ の二つのケースが調べられた。図 3.4 では実際のダイナミクスの様子および TTC の様子が示される。まずデルタ関数 $\delta(t)$ の直接投射に比べ、Non-Chaotic ESN を介して投射した条件において、全ての投射強度で一貫して TTC の値が高くなった。特にデルタ関数が直接投射される場合、どのような投射強度 u_{amp} に対しても TTC の値がほとんど変わらなかった。対照的に、Input Reservoir として Non-Chaotic ESN を介して入力が投射された時、投射強度 u_{amp} が大きくなるにつれ、TTC が向上した。この結果は、まず Input Reservoir で生成されるような過渡的な緩和過程の Innate Training における必要性を示唆する。また緩和過程が長い $g^{in} = 0.99$ の条件の方がより弱い投射強度で TTC の値が上がり始めた。これは $g^{in} = 0.99$ の方が $g^{in} = 0.10$ の時に比して、Chaotic ESN における入力同期を起こし局所的に ESP を高めやすいことに起因すると考えられる。また $g^{in} = 0.10, 0.99$ とともに TTC が一定の値に飽和し、若干低下し始める挙動が観察されたが、これは Innate Training において用いられる入力において、適切な強度の範囲の存在を示唆する。

3.3.2 ソフトボディを用いた実験

前項では Input Reservoir として ESN を使用した。冒頭で議論されたように、身体の柔軟な要素の Input Reservoir としての機能性を議論するため、ESN と同様に緩和過程を生じるソフトボディを Input Reservoir として採用する。本章では図 3.5 に示されるようなセンサが取り付

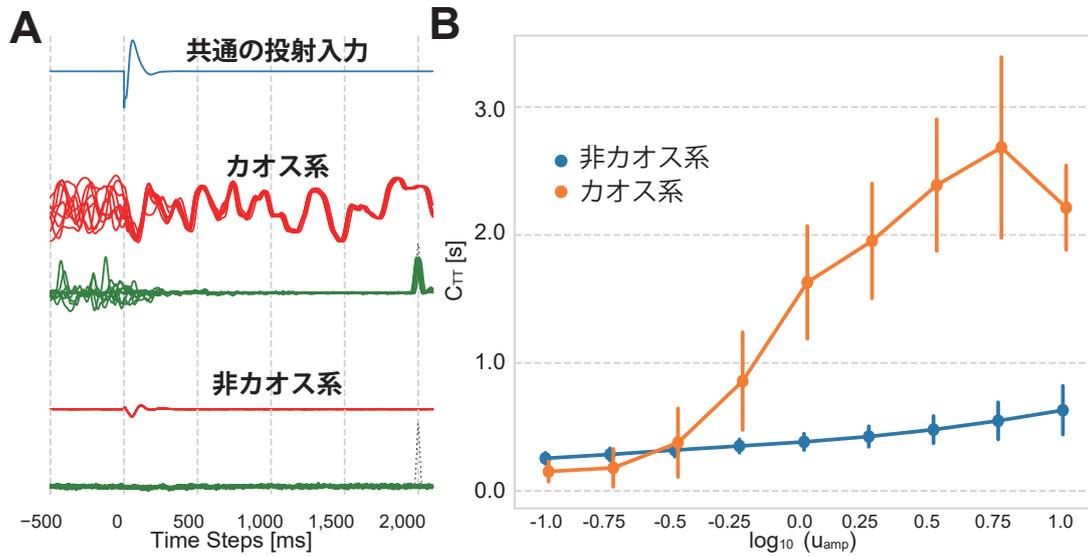


Fig. 3.3 Innate Training におけるカオス性の重要性の評価. 投射先として Chaotic ESN の代わりに Non-Chaotic ESN ($g = 0.9$) を使用し, Chaotic ESN を使った場合と比較した. (A) 実際の入力およびダイナミクスの様子. カオス系では複雑なダイナミクスが生成され続けるのに対し, 非カオス系では入力を投射してもダイナミクスが一定期間後に消失してしまう. (B) TTC の比較. 一定の強度以上でカオス系 ($g = 1.6$) が非カオス系 ($g = 0.9$) を大幅に上回る.

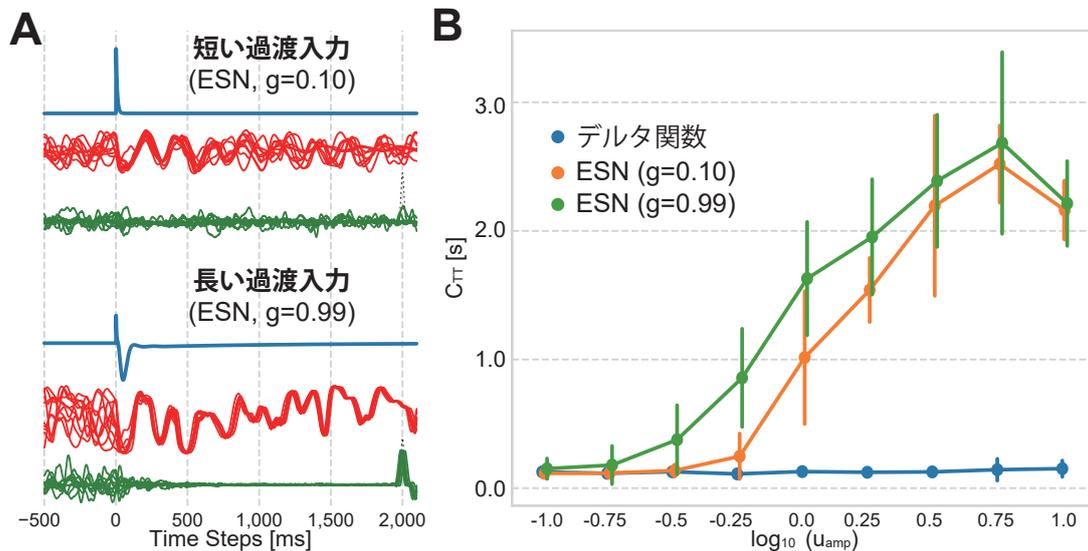


Fig. 3.4 Input Reservoir の性質が与える Innate Training の性能への影響. (A) ESN の内部ダイナミクスおよび Timer Task における出力例. Input Reservoir のスペクトル半径 g が大きいほど, 減衰の遅い長い過渡的入力が投射される. この例では短い過渡入力ではパルス波 (点線) の生成に失敗しているのに対し, 長い過渡入力では適切に生成されている. (B) TTC の比較. 長い過渡入力 ($g = 0.99$) を与えることで, 比較的弱い強度の領域でも C_{TT} が高くなる.

けられた，シリコン（Silicone）ゴム製のたこ足状のソフトボディ [69] を使用した．このソフトボディはサーボモータによって制御され，モータコマンド m によって運動角度を調整できる．実験では初期姿勢が $m = 0$ によって設定されたのち， $m > 0$ の値を指定することで動きを発生させる．この時モータコマンド m の値が大きいほど大きな運動が発生し，結果的に定常状態（今回のケースだと停止）に至るまで長い時間を要する．ソフトボディには 10 個のセンサが取り付けられており，10 次元のダイナミクス $\mathbf{x}^{\text{soft}}(t)$ が得られる．実験では各条件でサンプリングされたセンサ時系列を適切に補間し，Chaotic ESN に以下の式の形で投射された．*7

$$\mathbf{u}^{\text{ch}}(t) = U^{\text{sc}} \mathbf{x}^{\text{soft}}(t) \quad (3.6)$$

なお $U^{\text{sc}} \in \mathbb{R}^{N^{\text{ch}} \times 10}$ は一様乱数 $\mathcal{U}([-1, 1])$ よりサンプルされ生成された．図 3.6 では実際のセンサの値および，得られた TTC の結果が表示される．まず図 3.6A から示される通り，モータコマンドが大きい $m = 0.9$ の時に ESN と同様 Innate Training によって再現的にカオス軌道を生成させることに成功した．また図 3.6B から，長い緩和過程の方が弱い u_{amp} に対して TTC が上がりやすく，かつある一定強度でその値が飽和する様子が分かる．これは先ほどの ESN の結果と類似しており，ソフトボディで発生するような実際の緩和ダイナミクスも，カオス力学系に接続することで Input Reservoir として活用されることを示唆する．

3.4 議論

本章では，Innate Training を成立させるような過渡的入力を Non-Chaotic ESN を用いてモデル化した．また Input Reservoir と呼ばれる概念を導入し，ある大きな ESN の部分系として内在化させた．そしてこのように入力時系列の生成機構が内在化された ESN の設定においても，ある適切な投射強度において Innate Training が成立することを示した．したがって，第 2 章で指摘された過渡的入力の生成機構に関する非自己充足的な構成は解消されたといえよう．また ESN を用いモデル化し Input Reservoir として内在化させる着想は，今回の入力時系列のみならず Innate Training 内のあらゆる時系列に適用可能である．実際第 7 章では，教師軌道を生成する Chaotic ESN が Input Reservoir として RNN の中に内在化される．このように Input Reservoir という概念は，外的に付与される時系列データを内在化させる着想を与え，学習システムの一貫した表現の実現において重要な役割を果たす．

また本章では Non-Chaotic ESN として表現された過渡的入力の性質と，Innate Training の学習性能の関係が調査された．その結果，直接入力を投射した時に比べて，Non-Chaotic ESN を介して過渡的入力が投射される場合，TTC，すなわち時間的情報処理能力が大幅に向上することを示した（図 3.3）．また特にスペクトル半径 g^{in} が大きい，すなわちより長い緩和過程

*7 サンプリング周期が異なるので，適当に $\mathbf{x}^{\text{soft}}(t)$ の線形補完し，ESN との時間幅を揃えた．

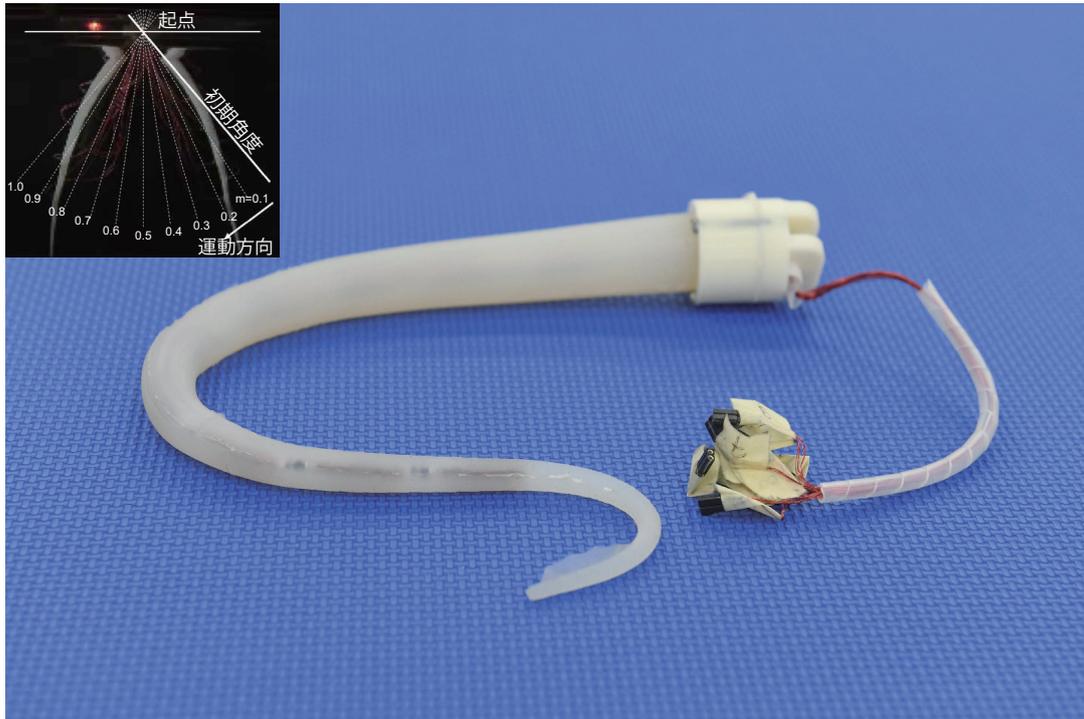


Fig. 3.5 使用されたソフトボディおよびその制御方法. 本実験では写真のようなセンサが取り付けられた、たこ足状のソフトボディを使用した. このソフトボディはサーボモータに与えられるモータコマンド m により制御される. 左上の図に示されるようにモータコマンドにより, サーボモータの角度が変わりソフトボディの運動が発生する. 特に大きな m の値ほど大きな運動が発生し, そのセンサダイナミクスの緩和過程は長くなる.

が生成される時, 比較的投射強度が低い場合でもその時間的情報処理能力が高くなりやすいことを示した (図 3.4). これらの結果から長い時定数を持つ過渡的入力, **Innate Training** の事前学習において重要であることが示唆されるとき. 一方で投射強度を大きくした時, スペクトル半径の大きさにかかわらず, **TTC** の値がある一定の値に飽和した. これは **Innate Training** の事前学習において, ある適切な範囲の投射強度が存在することを示唆する. またこの **TTC** の飽和値は系の大きさ, すなわち **Chaotic ESN** の次元数によって規定されていると推測される. この系の大きさと時間的情報能力の関係は第 4 章においてより詳細に解析される.

また本章ではさらに一步踏み込んで **Input Reservoir** の概念を適用し, 生物の身体が持つ柔軟な要素の機能性を検証した. 脳・神経系は一般に皮膚や筋肉における受容器を介して環境の情報を受け取るが, その受容器は身体と密に結合しており常にその身体の影響を受けていると考えられる. また生物の身体はしばしば多くの柔軟な要素を含み, それらの要素は環境入力に対するフィルタとして機能している. 本章で導入された **Input Reservoir** を用いて, そのような生物において広く観察される構成の機能性をも, 計算論的に評価することが可能となった. 本章では **Innate Training** の学習性能への寄与のみが検証されたが, 無論他の力学系的な指標を

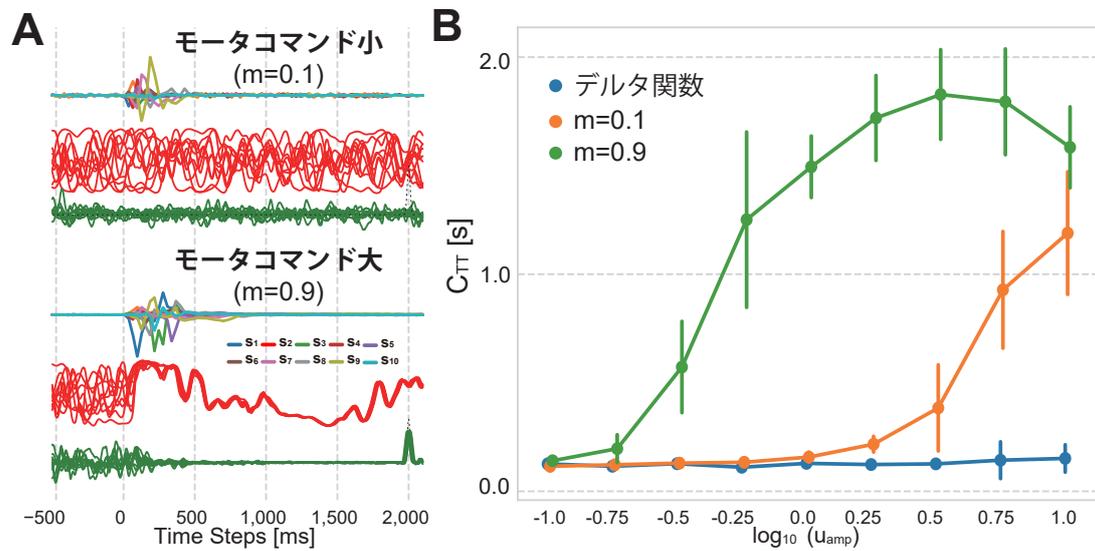


Fig. 3.6 ソフトセンサを用いた Innate Training の評価. (A) 投射されるセンサダイナミクスおよび, Timer Task における出力例. モータコマンドが大きいほど振幅が大きく, 結果的に減衰の遅い長い過渡的センサ値のダイナミクスが投射される. この例では $m = 0.1$ ではパルス波 (点線) の生成に失敗しているのに対し, 長い過渡ダイナミクスを生じる $m = 0.9$ では適切に生成されている. (B) TTC の比較.

導入して検証することが可能であり, 今後それらが系統的に解析されることが期待される.

第 4 章

高次元カオスを活用した自発的な記号の切り替え

本章では、Innate Training では設計者によって外的に制御される入力切り替えを、学習システムによって自発的に制御させる構成を実現する。特に Input Reservoir ならびに高次元カオス力学系が生成するカオス軌道を活用し、線形な閉ループを介して高い操作性で自発的な遷移則を自在に実現できる手法 [78] を提案する。そしてデモンストレーションと数値解析を通して、提案手法の有効性と重要性を示す。また関連する重要な話題として、カオス的遍歴の軌道設計としての本提案手法の妥当性を議論する。

4.1 問題設定

4.1.1 自発的な記号の切り替え

Innate Training を俯瞰した時、設計者が入力切り替えを制御する構成が確認できる。すなわち「ある入力に対しての内部のカオス的応答を再現的にさせる」手法が Innate Training であるが、その「ある入力」の生成や選択は設計者に依存している。そしてその設計者の選択は無論学習システムとは独立している。一方で第 3 章で Input Reservoir と呼ばれる概念を導入し、過渡的入力の生成機構を ESN において自在に内在化できるようになった。言い換えれば入力時系列の生成は、Input Reservoir として部分系を導入することで同一の ESN 内で表現されるようになった。したがって、ここで問題になるのは、現在は設計者の手に委ねられている入力の種類の選択の方である。このような入力の種類の選択が、学習システム自身によって設計者から独立して自発的に達成されれば、第 1 章で議論された、学習システムの自発性（要件 1）ならびに学習対象の自己決定・構成能（要件 2）をより向上させることができる。^{*1}

*1 一方でこの外的な入力切り替えは学習システムに対する一種の教示とも解釈でき、その場合必ずしも自発的

さて入力を選択という操作に関して、無論入力のレパートリは無数に存在するため、無数の選択肢が存在する。ただ実際には無数の入力全てに対して学習されることはないので、問題を簡単にするため有限個の入力の種類、すなわち記号がある状況を考える。^{*2}Innate Training の典型的な設定では設計者が、例えば「入力 A を 2000 ms 与えたのち、入力 B に切り替えそれを 4000 ms 与える操作を繰り返す」のような所定のルールの下で、所定の記号の切り替えを制御している。^{*3}したがって、自律的な学習システムでは、このような記号の切り替えに関するルール、すなわち記号遷移則を学習システム自身が自発的に実現することが望ましい。そこで本章では、そのような自発的な記号の切り替えを行う高次元非線形力学系の構成を目指す。

4.1.2 系が満たすべき制約

さて第 1 章で定義された自律的な学習機能の 3 要件に基づき、入力の切り替えに関して以下の 3 点の性質を、構成される高次元非線形力学系が満たすべき制約として導入する。

制約 1 構成される高次元非線形力学系の単一性

制約 2 同じ入力維持される期間が十分に長くかつ再現的

制約 3 設計者からの予測不能性

まず階層・モジュールなしに、単一の高次元非線形力学系の上で構成されることが望ましい。これは第 1 章で議論された自己充足性（要件 3）に関する制約である。例えば記号の生成とその間の遷移を制御する機構が分けられた階層的な構成を導入することで、このような遷移則は比較的容易に実装される。しかしながらそのような構成は自己充足的でないばかりか、ある限定された記号遷移則に特化され、適応性が損なわれる。また生物知能ではこのような静的な構造は最初から存在せず、むしろ身体の発達や環境との動的な相互作用を通じて構成されると考えられる。したがって、ここでは、単一の高次元非線形力学系のダイナミクスの上で所望の記号遷移則（特に階層的な構造が見いだされるダイナミクス）を実現することを目指す。

次に入力の切り替えに関して、入力が一定期間保持され続けられなければならない。この制約は Innate Training の安定性に関する制約、すなわち学習機能の恒常性のために導入される。Innate Training では各エポックにおいて教師軌道を L [ms] にわたって調整する操作が行われ

に構成される必要はない。したがって、ここでは追加可能な構成として、設計者からの受動性を完全に無にするのではなく軽減する手法の開発が目論まれる。

^{*2} 無論無数のレパートリの中から有限個の選択肢を抽出するのは設計者であり、それは要件 2 に抵触する。一方でその抽出を自律的に実現するには、分節化や言語獲得といった形で記号を学習システム自体に生成させる必要があり、より問題が複雑化する。ここでは Innate Training そのものを高次元非線形力学系のダイナミクスで表現する方法論が扱われるため、その記号生成の問題には踏み込まずあえて問題を限定化する。この記号の生成に関しては、今後の展望として第 8 章において議論される。

^{*3} Innate Training [47] では同じ入力を繰り返し与えていたが、これも一種の記号的な操作として捉えられる。

る。この L は ESN の時定数に比して非常に大きい値が一般に採用される。^{*4}これは換言すれば、入力の切り替えに関して、系が十分に長い Timer Task を解くことが要求されていることを意味する。

最後に、その記号遷移則に関して、設計者から予測不能であることが望ましい。これは自発性（要件 1）に関する制約で、設計者からの独立性を担保するために導入される。このような設計者からの予測不能性を実装する手法として、例えばマルコフモデルのような確率モデルが候補として挙げられるだろう。しかしながら第 1 章で議論されたように、確率モデルでは乱数生成の機構が系のダイナミクスから完全に独立しているため、階層的構造が必然的に導入される。したがって、自己充足性（要件 3）が担保された構成を実現するには、確率的な要素を含まずに実装されることが望ましい。反面、力学系のカオス性を活用することで、このような階層的構造の導入なしに、一貫した高次元非線形力学系の上で自発的な記号遷移則を実現できる。そこで本章では高次元カオス力学系のモデルとして Chaotic ESN を用い、その上で自発的な記号遷移則を実現する手法を提案する。

4.1.3 関連研究

ESN, もっと一般的には RNN を用いてこのような記号力学（記号の切り替え）を制御する手法はこれまでいくつか提案されている。しかしながらそれぞれ先述の制約に抵触する要素を含む。例えば Namikawa と Tani [79, 80, 81] は人型ロボットを用いて、モーションプリミティブ（Motion Primitive）間の確率的な切り替わりで特徴づけられる階層的な行動を、RNN を用いた決定論的な制御器を用いて擬似的に実現した。^{*5}しかしこの制御器では、各モーションプリミティブに応じてそれぞれ複数の小さい時定数の RNN モジュールが用意され、それらを大きい時定数の ESN がその出力を切り替えるような階層的な構造が制御器において先行的に採用されている。先述の通り、このような構成は学習システムの自己充足性を損ない、本研究が目論む自律的な学習システムには適さない。また BPTT 法を用いて内部のパラメータが全て調整されているが、長い時定数の入出力関数の RNN への埋め込みにおいて、勾配降下法は一般に不安定であり、また多数の学習を必要とする [82]。他にも Yamashita と Tani [83] の方法では記号に対応したモジュールは排除されているが、依然として時定数の違いで表現される明示的な階層が導入されており自己充足的でない。また連想記憶モデルに基づく方法 [84, 85] が提案されておりその系はカオス的ではあるものの、埋め込まれる大域的な遷移則を制御することは困難である。また軌道アトラクタへの引き込み現象を活用してロボットの制御を実現する手法 [86] が提案されているが、そこでは埋め込まれる軌道がリミットサイクル状の周期的なものに限定され、それ単体で自発的な遷移を表現することは困難である。

^{*4} 例えば [47] では、時定数 $\tau = 10$ ms の Leaky ESN に対し、 $L = 1000 \sim 3000$ ms の程度の値が設定される。

^{*5} 擬似的、と書いたのは決定論的であるために、本来それは確率的とはいえないからである。

4.1.4 手法の指針

そこで本論文では、階層を持たない Chaotic ESN を活用し、所望の記号遷移則を確率的なものも含め設計できる手法を提案する。提案手法は3段階からなるバッチ学習である（図4.1）。

- 高次元カオス力学系を用意し、離散的な入力（記号）に応じて独立な軌道が再現的に生成させるように Chaotic ESN の内部結合を調整する。特に階層的な構造^{*6}やモジュールが含まれないネットワークの構成が採用される。また、ESN の記号入力に関する時間的情報処理能力が向上し、記号の切り替えをオフセットとする長い Timer Task を解けるようになり、二つ目の安定性に関する制約が満足されることが期待される。同時に再現的に生成される高次元カオス軌道を活用し、所望の軌道（出力軌道）を出力させるように線形な変換器（Readout）を調整する（Step 1）。本手法は力学系のモジュール性や階層性が必要とされないため、より広い範囲のカオス力学系に適用可能である。たとえば物理系 [51] やそこで現れる高次元カオスを、本提案手法を用いて記号遷移則の埋め込みに活用することができる。また相互作用および Readout の学習は、RC の手法を活用し、少数自由度の部分的なパラメータの調整のみで達成される。この学習は、誤差逆伝搬法を用いずに達成されるため、従来の手法に比べて安定的かつ計算量的に低コストである。
- 線形な閉ループ識別器を Step 1 で学習された高次元カオス力学系に追加して、特定の記号力学を設計者なしに生成させる（Step 2）。ここでは ESN の内部結合および、Readout のパラメータは固定化される。この時閉ループ識別器は Step 1 の Readout と同じく、ゲート機構のような階層的な構造を必要としない。すなわち生成されるカオス軌道の高次元性と分離性能を最大限活用することで、必要最小限の計算能力^{*7}の付与で特定の記号遷移則が設計される。
- Step 2 で加えられた線形な閉ループを調整して、確率的な記号遷移則を設計する（Step 3）。特に力学系に内在するカオス性を活用して、決定論的な力学系において疑似的に確率遷移則を模倣させる。この時点で先述の3点の制約を満たした、自発的に記号が切り替えられる高次元カオス力学系が構成されたといえる。

*6 異なる時定数の力学系。

*7 すなわち非線形性および記憶性能。

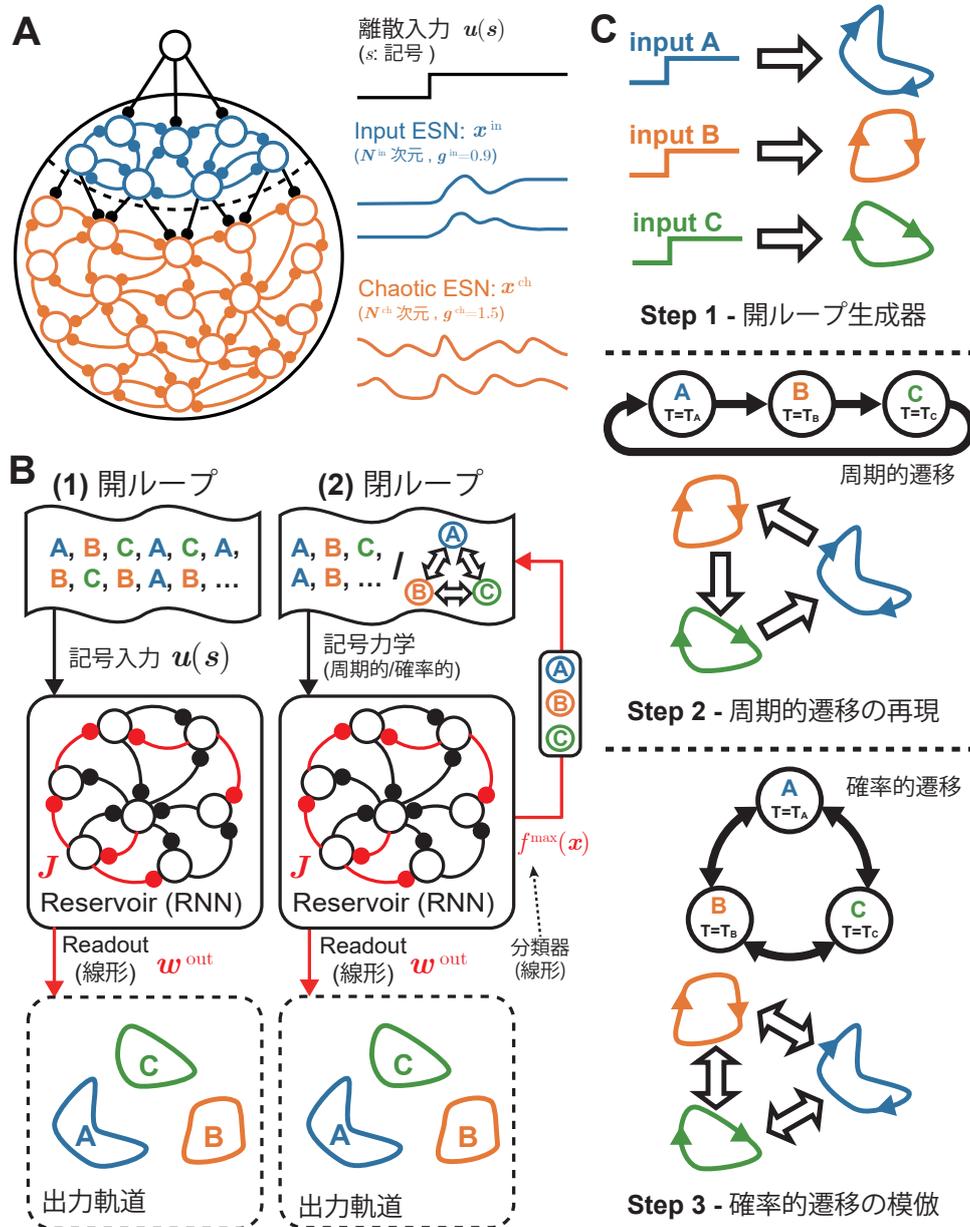


Fig. 4.1 実験系の模式図. (A) 本実験で用いられる高次元力学系のモデル. 特に Input ESN と Chaotic ESN に分けられる ESN が使用される. Input ESN は離散的な入力と Chaotic ESN の間のフィルタ, すなわち Input Reservoir としての役割を持ち, 記号入力切り替わる時に Chaotic ESN に投射する過渡的入力生成する. (B) 二つの実験設定. 開ループ系では記号入力が外的に与えられる. 一方で閉ループ系では記号入力が系自身によって生成される. 図中で赤色で表現される要素が調整される. (C) 本論文で提案される3段階からなるバッチ学習の概要図. Step 1 では記号入力に応じて, 高次元カオス軌道および出力軌道を生成するように, ESN の内部結合および Readout が学習される. Step 2 および Step 3 では記号入力が系自身によって生成させる. 特に Step 2 では周期的な記号遷移則, Step 3 では確率的な記号遷移則の埋め込みが目論まれる.

4.1.5 カオスの遍歴と自発的な記号遷移

自発的な記号遷移と関連する重要な概念として、カオスの遍歴 [87, 88, 89] がある。カオスの遍歴は高次元力学系上で観察される非線形現象で、複数の局所的に安定な領域（擬似アトラクタと呼称される）の間の、不規則かつカオス的な遷移によって特徴づけられる。一般にカオス力学系は初期値鋭敏性を有し、初期状態の差分が時間発展によってある方向において指数的に拡大される。一方でカオスの遍歴を生じる力学系は、大域的にはカオス的にもかかわらず、局所的に予測可能なダイナミクス、すなわち疑似アトラクタを繰り返し生成する。そしてカオスの遍歴において興味深いのはこのような階層的なダイナミクスが、階層的な構造を持たない力学系から発生する点である。言い換えればカオスの遍歴という現象の存在は、明示的な階層を系が内包せずとも、階層的に見えるダイナミクスを生じうることを暗示する。本提案手法により ESN 上で記号間の自発的な遷移が達成された場合、記号と擬似アトラクタが対応付けられることで、一種のカオスの遍歴が達成されたと解釈できる。

4.2 提案手法

以下実験設定に関して詳細に説明される。

4.2.1 Chaotic ESN を基盤とする実験系

本手法では、疑似アトラクタに対応する M 種類の離散的な記号の $s \in S$ ($S := \{s_1, s_2, \dots, s_M\}$) が予め用意される。すなわち M 種類の記号と、その M 種類の記号間の遷移則の ESN における埋め込みを目指す。特に図 4.1A において示されるように、記号 s に対応する離散的な入力が直接投射される Input ESN (N^{in} 次元) と、カオス軌道を生成する Chaotic ESN (N^{ch} 次元) から構成される ESN を使用する。これは第 3 章で提案された Input Reservoir によって、入力の生成機構が内在化された ESN である。Input ESN のダイナミクス $\mathbf{x}^{\text{in}}(t) \in \mathbb{R}^{N^{\text{in}}}$ および、Chaotic ESN のダイナミクス $\mathbf{x}^{\text{ch}}(t) \in \mathbb{R}^{N^{\text{ch}}}$ は以下の式で表現される。

$$\tau \frac{d\mathbf{x}^{\text{in}}}{dt}(t) = -\mathbf{x}^{\text{in}}(t) + \tanh\left(g^{\text{in}} J^{\text{in}} \mathbf{x}^{\text{in}}(t) + \mathbf{u}^{\text{in}}(s(t))\right) \quad (4.1)$$

$$\tau \frac{d\mathbf{x}^{\text{ch}}}{dt}(t) = -\mathbf{x}^{\text{ch}}(t) + \tanh\left(g^{\text{ch}} J^{\text{ch}} \mathbf{x}^{\text{ch}}(t) + J^{\text{ic}} \mathbf{x}^{\text{in}}(t)\right) \quad (4.2)$$

ここで $\tau \in \mathbb{R}$ は時定数、 $g^{\text{in}}, g^{\text{ch}} \in \mathbb{R}$ は非線形性を制御するパラメータ、 $\mathbf{u}^{\text{in}}(s) \in \mathbb{R}^{N^{\text{in}}}$ は記号入力 s が与えられた時、Input ESN に投射される入力、 $J^{\text{in}} \in \mathbb{R}^{N^{\text{in}} \times N^{\text{in}}}$ 、 $J^{\text{ch}} \in \mathbb{R}^{N^{\text{ch}} \times N^{\text{ch}}}$ は結合行列、 $J^{\text{ic}} \in \mathbb{R}^{N^{\text{ch}} \times N^{\text{in}}}$ は Input ESN と Chaotic ESN をつなぐフィードフォワードな結合行列を意味する。

J^{in} の各要素は正規分布 $\mathcal{N}\left(0, \frac{1}{N^{\text{in}}}\right)$ よりサンプルされる．一方で J^{ch} は密度 $p = 0.1$ のランダムスパース行列であり，各要素が正規分布 $\mathcal{N}\left(0, \frac{1}{pN^{\text{ch}}}\right)$ よりサンプルされる．Input ESN の非カオス性と Chaotic ESN のカオス性を保証するため $\tau = 10.0, g^{\text{in}} = 0.9, g^{\text{ch}} = 1.5$ を採用した [90]．またバイアス項によって引き起こされる分岐によって Chaotic ESN のカオス性が抑圧される [68] ことを防ぐため，同じ記号入力を与えられ続けた時に， $J^{\text{ic}} \mathbf{x}^{\text{in}}(t)$ が 0 に収束するように J^{ic} を予め設定する．すなわち第 3 章で議論した過渡的な入力を投射するように設計されている．*8

これらをまとめれば，式 (4.1) and (4.2) を統合した ESN 全体のダイナミクスは $\mathbf{x}(t) \in \mathbb{R}^{N^{\text{in}}+N^{\text{ch}}}$ 以下の単一の式で表現される（ただし \odot は要素積を表す）．

$$\tau \frac{d\mathbf{x}}{dt}(t) = -\mathbf{x}(t) + \tanh(\mathbf{g} \odot (J\mathbf{x}(t)) + \mathbf{u}(s(t))) \quad (4.3)$$

ここで $\mathbf{x}, \mathbf{g}, J, \mathbf{u}$ は以下の式より定義される．

$$\mathbf{x}(t) := [\mathbf{x}^{\text{in}}(t); \mathbf{x}^{\text{ch}}(t)] \quad (4.4)$$

$$\mathbf{g} := \underbrace{[g^{\text{in}}, \dots, g^{\text{in}}]}_{N^{\text{in}}} \underbrace{[g^{\text{ch}}, \dots, g^{\text{ch}}]}_{N^{\text{ch}}}^T \quad (4.5)$$

$$J := \begin{bmatrix} J^{\text{in}} & \mathbf{0} \\ J^{\text{ic}} & J^{\text{ch}} \end{bmatrix} \quad (4.6)$$

$$\mathbf{u}(s) := [\mathbf{u}^{\text{in}}(s); \mathbf{0}] \quad (4.7)$$

一方で出力軌道は，ESN の内部状態 \mathbf{x} の線形変換で構成される．すなわち Readout $\mathbf{w}_{\text{out}} \in \mathbb{R}^{N^{\text{in}}+N^{\text{ch}}}$ が所望の目標関数 $f_{\text{out}}(t)$ に関して以下の式を満たすように学習される．

$$\mathbf{w}_{\text{out}}^T \mathbf{x}(t) \approx f_{\text{out}}(t) \quad (4.8)$$

また最終的には外的に与えられる記号列 $s(t)$ そのものが，図 4.1B-2 に示される閉ループによって生成される．この時，閉ループとして，以下の式で表される分類器 $f_{\text{max}} : \mathbb{R}^{N^{\text{in}}+N^{\text{ch}}} \rightarrow \mathcal{S}$ を取り付ける．

$$f_{\text{max}}(\mathbf{x}(t)) := \underset{s \in \mathcal{S}}{\operatorname{argmax}} \mathbf{w}_s^T \mathbf{x}(t) \quad (4.9)$$

ここで $\mathbf{w}_s \in \mathbb{R}^{N^{\text{in}}+N^{\text{ch}}}$ は結合行列で，特定の記号力学 $s(t)$ を実現するように学習させる．*9まとめると本手法では，疑似アトラクタを生成する ESN の内部結合 J ，出力軌道 f_{out} を生成する Readout \mathbf{w}_{out} ，および記号力学 $s(t)$ を生成する \mathbf{w}_s を調整することにより，所望の疑似アトラクタ，出力軌道および記号力学が設計される．

*8 詳細な設定は付録 B を参照せよ．

*9 すなわち $s(t + \Delta t) \approx f_{\text{max}}(\mathbf{x}(t))$ ．ただし $s(t + \Delta t)$ は次の離散時間の入力， Δt は時間幅を表す．

そこで、第2章で述べられた FORCE 学習および Innate Training を活用し、自発的な記号切り替えを高次元カオス力学系上に構成する汎用的な手法を提案する。本提案手法は以下の3段階から構成されるバッチ学習である (図 4.1C)。

4.2.2 Innate Training による高次元カオス軌道的设计 (Step 1)

Step 1 では、記号に応じた高次元カオスを設計するため、Innate Training によって Chaotic ESN の内部結合 J^{ch} が調整される。まず M 種類の記号 s に対して、教師軌道 $\mathbf{x}_{\text{target}}^s(t)$ を記録する。ここで $\mathbf{x}_{\text{target}}^s(t)$ は $t=0$ において記号 s に切り替わった時の、ある初期値 $\mathbf{x}_{\text{target}}^s(0)$ と初期結合 J^{init} における軌道である。^{*10}Step 1 では、以下の目的関数 $C_{1\text{-in}}$ を最小化するように、 J^{ch} が Innate Training によって調整される。

$$C_{1\text{-in}} := \sum_{s \in S} \sum_{t=0}^{L_{\text{innate}}} \|\mathbf{x}^s(t) - \mathbf{x}_{\text{target}}^s(t)\|^2 \quad (4.10)$$

ここで、 $\mathbf{x}^s(t)$ は $t=0$ において記号 s に切り替わった時の軌道、 L_{innate} は学習される教師軌道の長さである。学習では Chaotic ESN のノードの内半分 ($N^{\text{ch}}/2$) が選択され、そのノードに接続する結合パラメータが J^{ch} の中で調整される。学習は各 s に対して 200 エポック行われ、 $C_{1\text{-in}}$ が最小になる J^{ch} を採用する。^{*11}この Step 1 における Innate Training の後、記号入力が s に切り替わった後に ESN が $\mathbf{x}_{\text{target}}^s$ を $L_{\text{innate}}[\text{ms}]$ にわたり再現的に生成することが期待される。^{*12}

同様に出力軌道を出力させるように \mathbf{w}_{out} を学習する。記号入力 s に対応する出力軌道 $f^s(t)$ に関して、以下の目的関数を最小化するように \mathbf{w}_{out} を学習する。

$$C_{1\text{-out}} := \sum_{s \in S} \sum_{t=0}^{L_{\text{out}}} \|f^s(t) - \mathbf{w}_{\text{out}}^T \mathbf{x}^s(t)\|^2 \quad (4.11)$$

Innate Training で埋め込まれた局所的に安定なカオス軌道を活用して、この目的関数によって学習された Readout も安定的に出力軌道を生成すると考えられる。なお内部ダイナミクス $\mathbf{x}^s(t)$ を記録し Ridge 回帰によって学習される。^{*13}

^{*10} 以下の簡単のために、Step 1 においては入力が切り替わる時刻を $t=0$ に設定する。つまり実際には入力は任意の時刻で切り替わることに注意せよ。

^{*11} Step 1 の詳細なアルゴリズムは付録 B を参照せよ。

^{*12} なおこの教師軌道の初期値の選択に関して、特に重要な基準は存在しない。しかしながら異なる高次元カオスが生成され、結果的にそれが分離性能を向上につながるため、 $\mathbf{x}_{\text{target}}^s(0)$ 間の距離が大きいほうが好ましい。したがって、本章では教師軌道の初期値 $\mathbf{x}_{\text{target}}^s(0)$ を相空間からランダムに選択した。

^{*13} ここで L_{innate} が必ずしも L_{out} と一致しないことに注意されたい。すなわち L_{out} は L_{innate} より大きくなりうる。

4.2.3 系自身による周期的な記号の切り替え (Step 2)

Step 2 では、記号入力を自律的に生成させるために、線形な閉ループ f_{\max} を設計する。特に、Step 2 では T [ms] 毎に切り替わる特定の周期的な記号列を ESN 自身に生成させる。まず周期的な記号力学 $s_{\text{per}}(t)$ を生成する。そして開ループ系 (図 4.1B-1) で $s_{\text{per}}(t)$ を $T_{\text{rec}} := 500000\text{ms}$ にわたり与え、その時の内部ダイナミクスから $\mathbf{x}(t)$ を記録する。これらの記録されたデータセットを用いて $\mathbf{x}(t)$ から $s_{\text{per}}(t)$ を再構成するように f_{\max} を学習させる。 f_{\max} を構成する \mathbf{w}_s は以下の目的関数 C_2 を最小化するように最適化される。

$$C_2 := - \sum_{s \in S} \sum_0^{T_{\text{rec}}} \mathbb{1} \{s_{\text{per}}(t + \Delta t) = s\} \log \frac{e^{\mathbf{w}_s^T \mathbf{x}(t)}}{\sum_{k \in S} e^{\mathbf{w}_k^T \mathbf{x}(t)}} \quad (4.12)$$

なお最適化アルゴリズムとして Limited-Memory BFGS (Broyden Fletcher Goldfarb Shanno) アルゴリズム [91] を使用した。この最適化タスクは第 3 章で登場した Timer Task の一種であると考えられる。このモデルでは学習された f_{\max} が、埋め込まれた高次元カオス軌道を活用し、ある一定の時間遅れで記号を切り替えることが期待される。

4.2.4 カオス性を活用した確率的な記号切り替え (Step 3)

Step 3 ではある有限状態機械で表現される確率的な記号遷移を、 f_{\max} を調整して実現する。Step 2 で議論されたように、系が決定論的なのでこのような確率的な記号遷移則が埋め込まれるには、系のカオス性を用いて疑似的に模倣されることが期待される。学習の手順は Step 2 と同じである。すなわち、ある確率的な記号力学 $s_{\text{sto}}(t)$ を用意し、開ループ系で得られた 500000 ms の学習データ ($\mathbf{x}(t), s_{\text{sto}}(t)$) を用いて、 f_{\max} を学習させる。この時、 \mathbf{w}_s は以下の目的関数 C_3 を採用する。

$$C_3 := - \sum_{s \in S} \sum_0^{T_{\text{rec}}} \mathbb{1} \{s_{\text{sto}}(t + \Delta t) = s\} \log \frac{e^{\mathbf{w}_s^T \mathbf{x}(t)}}{\sum_{k \in S} e^{\mathbf{w}_k^T \mathbf{x}(t)}} \quad (4.13)$$

C_3 の最適化は C_2 と同様 Limited-memory BFGS によって達成される。^{*14}

4.3 結果

本節では、各段階におけるそれぞれのデモンストレーションと数値解析の結果を示す。

^{*14} ここで C_3 と C_2 の式の形が同じであることに注意されたい。すなわち、その目的関数の形式に目標とする軌道の性質は反映されていない。むしろ、この Step 2 と Step 3 はデータ依存であり、用意された記号列の性質が、得られる系のカオス性の有無を決定する。

4.3.1 記号に応じた高次元カオス軌道の安定的生成 (Step 1)

前述の通り Step 1 においてはまず記号の切り替え時に、記号に対応した教師軌道を再現的に生成するように Chaotic ESN の内部結合 J^{ch} が調整される。まず図 4.2A では例として、 $(M, L_{\text{innate}}) = (1, 1000)$ の条件で Innate Training された 1500 次元の ESN ($N^{\text{in}} = 500, N^{\text{ch}} = 1000$) の軌道が表示されている。Innate Training 前では、初期値の異なる軌道が 1000 ms 以前に拡散するのに対し、Innate Training 後では 1000 ms の期間（黄色で色付けされている領域）教師軌道 $\mathbf{x}_{\text{target}}^s(t)$ （点線）を再現的に生成するようになる。また興味深いことに、1000 ms の範囲を超えてしばらくの間も、複雑な軌道が高い再現性で生成された。^{*15}

次に図 4.2B では $(M, L_{\text{innate}}) = (3, 1000)$ の条件で Innate Training された 1500 次元の ESN ($N^{\text{in}} = 500, N^{\text{ch}} = 1000$) の軌道および出力軌道が示される。この例では、最初は入力を与えず、途中からランダムな時間幅で記号が切り替わる。また $L_{\text{out}} = 1500$ ms の条件で、3 記号 A, B, C の入力に対してそれぞれリサージュ (Lissajous) 曲線、@マーク、ローレンツアトラクタ (xz 平面への射影) を出力するように二次元出力の Readout を学習させた。^{*16} その結果、各記号遷移後にそれぞれ異なる複雑な時系列パターンを持った軌道が高い再現性で生成され、その記号入力の遷移に応じいずれの初期値の軌道に関しても一定期間の間、所望の時系列パターンが再現的に生成される様子が観察された。ここで Readout として同一の線形結合 \mathbf{w}_{out} が用いられる点に注意されたい。すなわちこれらの結果は、同一の線形モデルでも記号に応じてそれぞれ独立に所望の時系列パターンを出力できるほど、各高次元カオス軌道が豊潤な表現能力を有することを示唆する。またこのような出力軌道の設計は、ロボティクスの文脈においても有効に活用される。例えば [92] では、環境入力の変化に応じて周期的な運動パターンを生成する実ロボットが提案されているが、そのような動作を実現する制御器は、本提案手法を用いて計算量的に安価な方法で素早く構築される。このように本提案手法は、即応性と適応性が求められるロボティクスの文脈でも有効であると期待される。

さらに Innate Training の適用範囲およびその妥当性を詳細に検証する (図 4.3)。まず、入力記号の数 M と Innate Training の精度の関係性を調べた。学習の指標として以下の式で表される Normalized Mean Square Error (NMSE) を導入した。

$$NMSE := \frac{1}{M} \sum_{s \in S} \left\langle \frac{\sum_0^{L_{\text{innate}}} \|\mathbf{x}^s(t) - \mathbf{x}_{\text{target}}^s(t)\|^2}{\sum_0^{L_{\text{innate}}} \|\mathbf{x}_{\text{target}}^s(t)\|^2} \right\rangle \quad (4.14)$$

^{*15} これは Innate Training の学習自体は成功しなくても、Innate Training において企図される再現的なカオス軌道の埋め込みが達成されていることを意味する。この Innate Training の特性に関する議論は、第 6 章においてさらに展開される。

^{*16} なお、アトラクタが完全に再現されたわけではないことに注意されたい。あくまでその過渡ダイナミクスが部分的に再現されたのみである。

ここでブラケットは、各記号における試行平均を表し、実験では 10 サンプルより計算された。図 4.3A は異なる学習条件における Innate Training の性能を表す。その結果、記号入力が多くなるほど、また L_{innate} を長くするにつれ、教師軌道 $\mathbf{x}_{\text{target}}^s(t)$ との間の NMSE が大きくなる傾向が観察された。この結果は、Innate Training によって埋め込まれる教師軌道の数に上限があることを示唆する。また記号の数 $M = 1$ に限定して、Chaotic ESN のノード数 N^{ch} と Innate Training の精度の関係を調べた。その結果、Chaotic ESN のノード数 N^{ch} を増やすにつれ、より大きい L_{innate} に対しても低い NMSE で Innate Training の学習が達成された (図 4.3B)。これらの結果は、埋め込まれる教師軌道の長さの上限が、ESN の次元数によって規定されることを示唆する。特により高次元な設定において、より長い教師軌道が Innate Training によって埋め込むことが可能となる。

次に Innate Training による系の情報処理能力の変化を評価する。この実験でも Timer Task を使用し、学習後の ESN において、入力情報がどれほど長く保持されるかを評価した。図 4.4C, D は TTF と対応する TTC を表す。まず TTC の解析により、教師軌道の長さ L_{innate} が大きいほど、より高い性能が得られることが分かる。また、TTC が $L_{\text{innate}} = 5000$ ms 付近で飽和する様子が観察された。これらの結果は、 L_{innate} が長いほど系の時間的な情報処理能力が向上するが、ある L_{innate} を境にその計算能力が頭打ちになることを示唆する。

さらに、Innate Training が与える ESN のカオス性に対する影響を評価した。決定論的な力学系において確率的な遷移を実現するためには、系がカオス的で予測不能性が内包されることが必要となる。本実験ではまず、記号遷移後の軌道の分散の度合いを評価する局所 Lyapunov 指数 (Local Lyapunov Exponent, 以下 LLE) を計測した。また無入力時 ($\mathbf{u}(t) = \mathbf{0}$) での最大 Lyapunov 指数 (Maximum Lyapunov Exponent, 以下 MLE) を計測し、システム全体のカオス性を評価した。^{*17} まず LLE の解析結果 (図 4.3E) から、記号遷移後の不均一な軌道の拡大が観察された。すなわち、Input ESN から投射される過渡的入力によって長期間にわたって収縮的な領域^{*18}が維持される様子が観察された。そして特に $t \in [0, L_{\text{innate}}]$ の領域において軌道の穏やかな拡大が観察された。これらの結果は Innate Training によって、相空間において局所的に収縮的な構造が生成されたことを示唆する。また MLE の解析より、特に $N^{\text{ch}} = 1000$ の時、いずれの L_{innate} においても安定的に正の MLE が得られた。このことは特に ESN の次元数が大きい時に、系のカオス性が安定的に維持されることを示唆する。^{*19}

^{*17} LLE と MLE の計算方法の詳細なアルゴリズムは付録 B を参照せよ。

^{*18} LLE が負の領域、これはカオス的遍歴における疑似アトラクタの長さに対応する。

^{*19} なお L_{innate} が小さい時急激な MLE の上昇が観察されたが、これは J のスペクトル半径の拡大に起因すると考えられる。 J の詳細な解析を付録 B において記した。

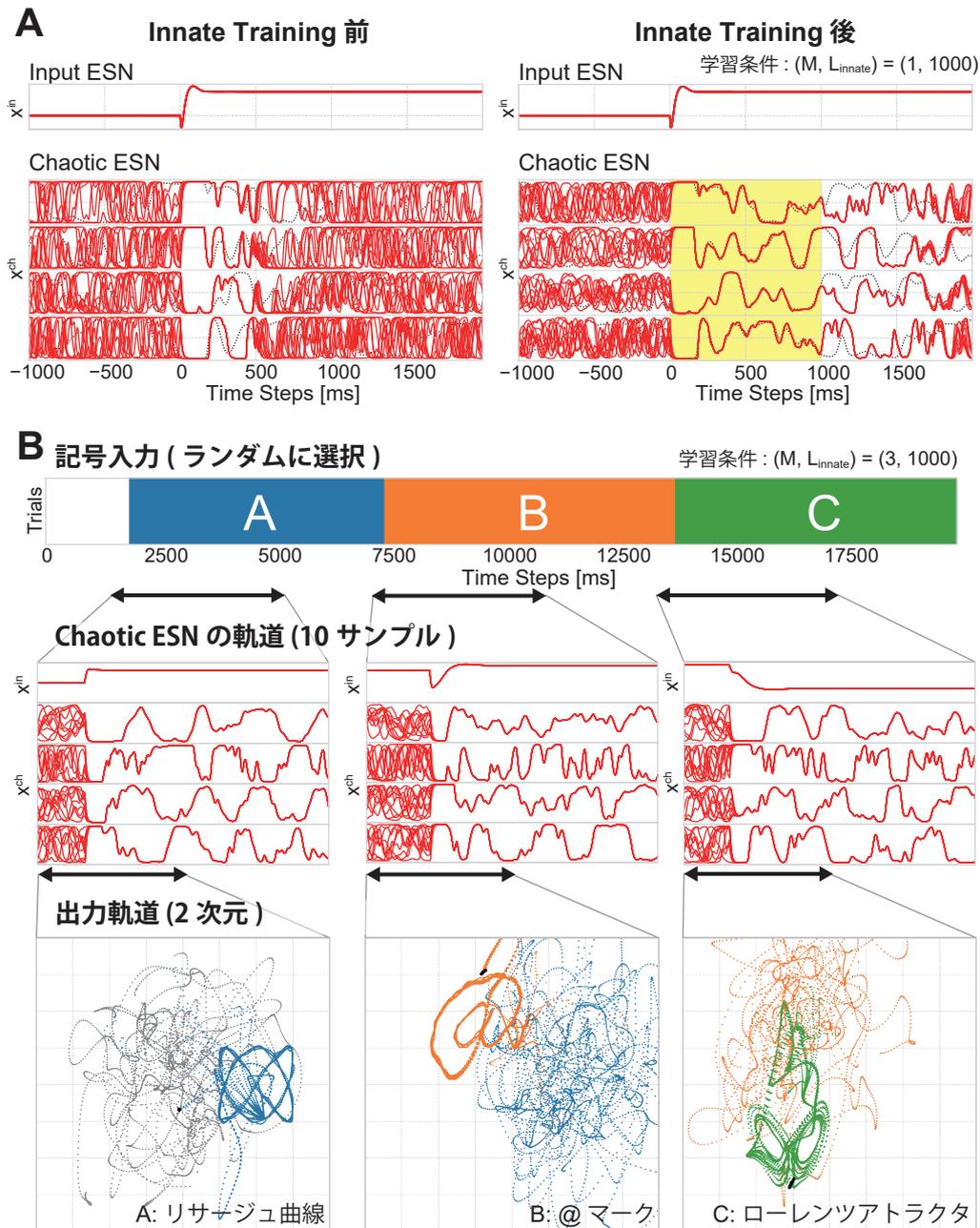


Fig. 4.2 Innate Training による複数の高次元カオス軌道の埋め込み (Step 1). (A) Innate Training 前後の ESN の軌道. グラフでは, $(M, L_{\text{innate}}) = (1, 1000)$ の条件で Innate Training を完了した ESN を用意した. 一番上の列の時系列データは Input ESN, 下 4 列の時系列データは Chaotic ESN の軌道を表す. 初期値の異なる 10 軌道 (赤) と教師軌道 $x_s^{\text{target}}(t)$ (点線) が重ね描かれている. (B) 開ループ系のデモンストレーション $(M, L_{\text{in}}, L_{\text{out}}) = (3, 1000, 1500)$ の条件で Innate Training が完了した ESN および Readout w_{out} を採用した. 記号入力の幅はランダムに選択された. このケースでは, 記号 A に対しリサージュ曲線, B に対し @ マーク, C に対しローレンツアトラクタの xz 座標を出力するように学習された Readout が使用された.

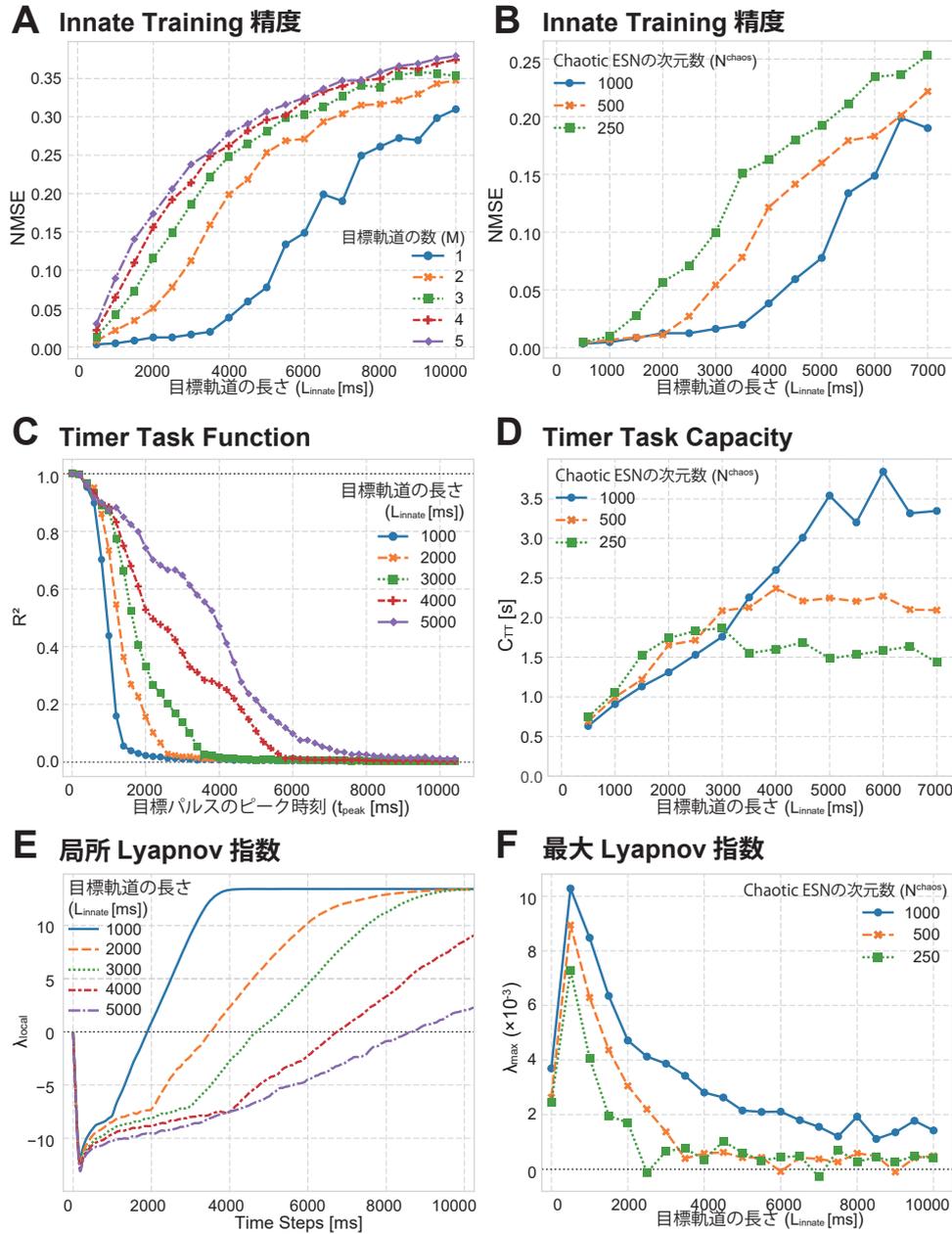


Fig. 4.3 Innate Training 後のシステムの性質の解析。(A) 記号の数 M と, Innate Training の性能の関係. NSME は 10 サンプルから計測される。(B) Chaotic ESN の次元数 N^{ch} と Innate Training の性能の関係。(C) TTF (Timer Task Function) の値. 10 回平均された値が表示される。(D) TTC (Timer Task Capacity) の値. TTC は TTF の積分値によって定義される。(E) 学習後の ESN の LLE (局所 Lyapunov 指数). LLE は Chaotic ESN に摂動を与えた時の時間発展より計算される (LLE の計算方法は付録 B を参照せよ)。(F) 学習後の ESN の MLE (最大 Lyapunov 指数).

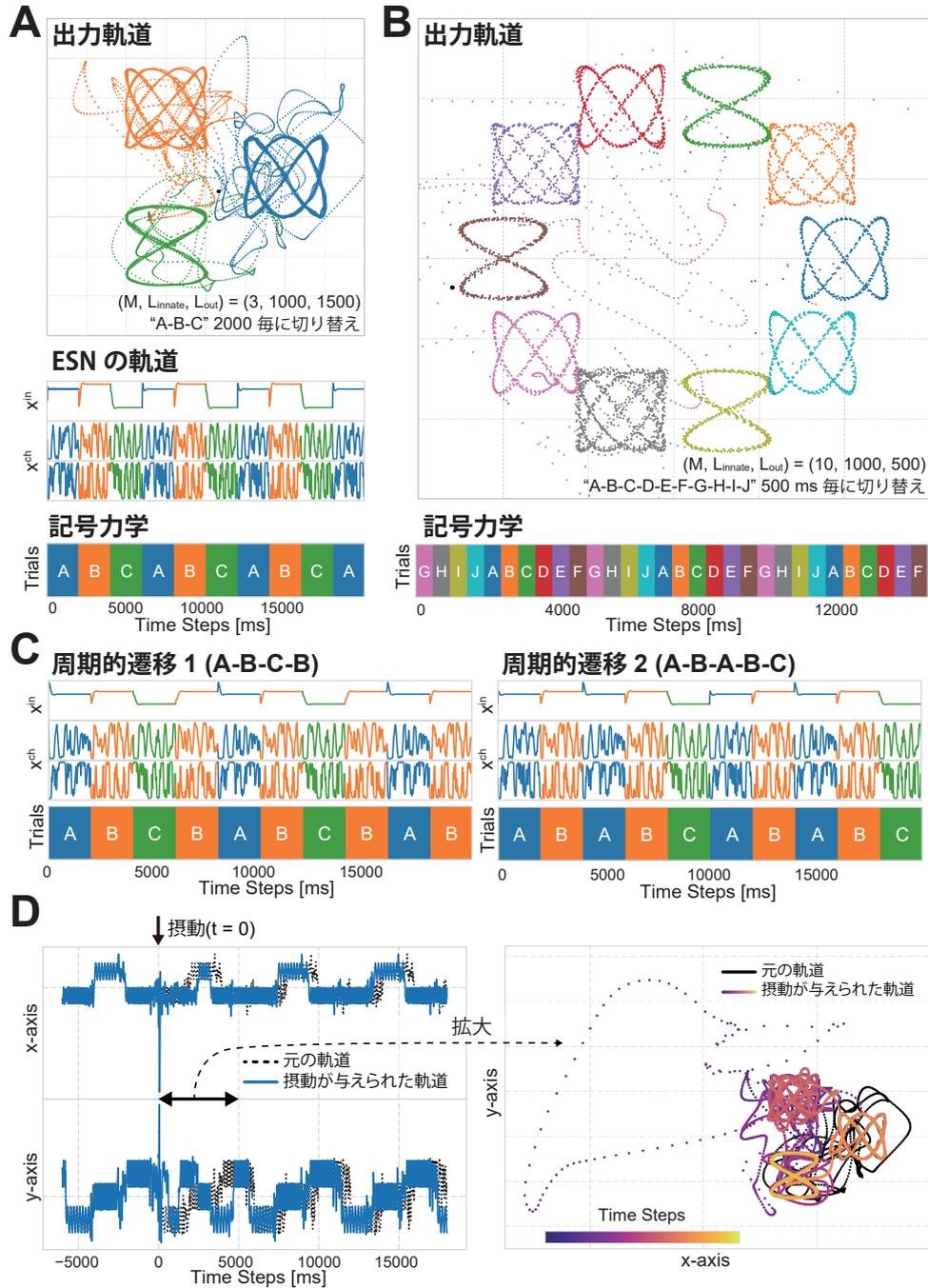


Fig. 4.4 閉ループ系による周期的な記号遷移の埋め込みのデモンストレーション (Step 2). (A) 3 記号周期の周期的な記号遷移. (B) 10 記号周期の周期的な記号遷移. (C) 履歴依存的な記号遷移の実現. (D) ESN に摂動を加えた時の出力軌道の挙動. $t = 0$ において微小な摂動が $x(t)$ に与えられた.

4.3.2 周期的な記号遷移の埋め込み (Step 2)

Step 2 では, Step 1 では外的に与えられた記号力学を ESN 自身で生成させる. 特に Step 2 では周期的な遷移を, 追加の線形な閉ループ f_{\max} によって実現する. さらに一定の時間幅で切り替わる様々な周期的な記号遷移が f_{\max} のみの調整によって自在に設計できることを示す. まず図 4.4A では $(M, L_{\text{innate}}) = (3, 1000)$ の条件で学習された ESN を用い, 時間幅 2000 ms の周期的記号遷移 A-B-C (周期 6000 ms) が実現される様子が示される. 同様に図 4.4B では $(M, L_{\text{innate}}) = (10, 500)$ の条件で学習された ESN を用い, 時間幅 500 ms の周期的記号遷移 A-B-C-D-E-F-G-H-I-J (周期 5000 ms) が出力された様子が示される. いずれのデモンストレーションにおいても, 所望の周期的な記号遷移が生成されるのみならず, 線形な Readout によって所定の出力軌道が安定的に生成されている.

また記号が重複するため, 高次の記憶性能が要求されるような周期的な記号遷移も, 同様の操作で容易に設計できることを示す. ここでは図 4.4A と同じ ESN を用い, f_{\max} のみを変えて A-B-C-B および A-B-C-B-A という記号遷移を学習させる. このような遷移は, 現在の出力だけでなく過去に出力した記号に応じて次に出力する記号を変えなければならないため, より難しいタスクといえる. 例えば A-B-C-B のような記号遷移の場合, 使用される記号の総数は A-B-C と遷移する形式と同じだが, B からの記号の切り替えにおいて, 前の記号に応じて次の出力が変化される必要がある. 実験では図 4.4A で用いられた ESN を再利用し, 閉ループ f_{\max} のみを学習させた. 図 4.4C に示される通り, ESN は A-B-C-B (8000 ms 周期) だけでなく, A-B-A-B-C (10000 ms 周期) の埋め込みに成功している様子が分かる. これらの結果は, 使用した学習済みの ESN が記号入力に対して, より高次の記憶性能を有することを示唆する. すなわち同じ記号入力に対しても, 以前の記号入力に応じて異なる軌道が生成され, それが本タスクを達成されるのに十分な分離性能を有することを意味する. ロボティクスの文脈では, よく Central Pattern Generator (CPG) のような非線形振動子を用いて生成されたりミットサイクルが制御に活用される [92, 93, 94, 95]. Step 2 で導入されたりミットサイクルの段階的な構成は, より長い周期を持つりミットサイクルの設計に活用されるだろう. 実際図 B.2 での解析において, 複数の遷移を含むような長周期の周期的アトラクタの埋め込みにおいて提案手法が FORCE 学習の性能を上回ることが示されている.*²⁰

さらに生成された軌道, および力学系の安定性を調べるため, 系に摂動を与えた時の出力軌道への影響を観察した. 図 4.4D では, $t = 0$ に $\mathbf{x}(t)$ 全体に弱い摂動が与えられた時の, 出力軌道の様子が示されている. その結果, 一度軌道を外れるものの, 元の周期軌道に一定時間後に戻る様子が観察された. またこの閉ループが追加された系に関して, MLE を計測したところ

*²⁰ 詳細は付録 B を参照せよ.

-1.89×10^{-4} となり極めて 0 に近い値が観察された。これらの結果は周期的な記号遷移が学習された閉ループ f_{innate} によって、系が非カオス的になったことを意味する。すなわち生成された内部ダイナミクス $\mathbf{x}(t)$ がリミットサイクルであることを示唆する。

4.3.3 確率的な記号遷移の埋め込み (Step 3)

Step 1 において高次元カオスを埋め込み、対応する出力軌道を構成した。Step 2 ではさまざまな周期的な記号遷移を f_{max} のみの調整で自在に系自身によって実現できることを示した。さて Step 3 では確率的な記号遷移を実現する。先ほど議論したように、決定論的な力学系で確率的な遷移を模倣するには、系のカオス性が活用される必要がある。

まず、 f_{max} の調整で確率的な遷移が疑似的に設計できることを示す (図 4.5A)。このデモンストレーションでは、図 4.4A で用いられた ESN および Readout を再利用する。そして 3000 ms の時間幅で記号 A, B, C が等確率に切り替わる記号遷移則を f_{max} の学習によって実現する。図 4.5B では、微小な初期状態からの記号力学、ESN のダイナミクス、および出力軌道が示される。グラフから示されるように、10000 ms 付近を境目に出力される記号遷移がばらつき始め、最終的に全く別の遷移パターンへ移行する様子が観察された。それにもかかわらずその出力軌道において、リサージュ曲線が安定的に生成されつづける様子が観察された。これらの結果は、大域的なカオス性によって擬似的に確率的な遷移が実現されつつも、Innate Training によって埋め込まれた高次元カオス軌道が記号遷移に応じて再現的に生成されることを意味する。^{*21}

さらに本手法の妥当性を詳細に解析するために、確率遷移行列および記号が維持される時間幅を計測した (図 4.5B)。実験では二種類の確率的な記号遷移則を用意した。まず遷移則 1 として図 4.5 と同じ有限状態機械を、遷移則 2 として遷移が一部限定された有限状態機械を採用した。また Step 2 の時と同じく、同じ学習済み ESN を使用し f_{max} のみを調整した。その結果、遷移確率および、切り替え時間において多少の誤差およびばらつきがあるものの、目標の確率遷移則と類似のパターンの埋め込みに成功した。また MLE を計測したところ、遷移則 1 のシステムにおいて $+2.01 \times 10^{-3}$ 、遷移則 2 において $+1.71 \times 10^{-3}$ の値が得られた。これらの結果は、系全体としては弱いながらもカオス性が維持されることを示唆する。実験ではさらに遷移の履歴依存性を詳細に解析した。その結果遷移確率には違いがそこまで反映されないものの、過去の記号に応じて出力軌道の分布の偏りが観察された。^{*22}その意味でこのシステムは確率的な遷移を、そのカオス性を活用してマクロ的^{*23}には実現している。

^{*21} 図 4.5A では単純かつ典型的な確率過程 (マルコフ過程) を対象としそのデモンストレーションを示したが、マルコフ過程等で表せない履歴依存的な遷移則も同等の手法で設計できる。そのデモンストレーションの詳細は付録 B および図 B.3 を参照せよ。

^{*22} 付録 B および図 B.4 を参照せよ。

^{*23} 記号遷移のレベルで、という意味。

最後に得られたカオス的なアトラクタおよび記号力学の構造を詳細に解析した。まずに Chaotic ESN に加えられた微小な摂動が記号力学に与える影響を観察した。その結果、遷移則 1, 2 が学習されたいずれの閉ループ系においても、ある一定時間後に記号のぼらつきが観察された (図 4.6A)。また、アトラクタの時間発展による終端状態の構造を解析するため、適当に選択された二つの次元によって張られる平面の時間発展、および対応する記号力学を観察した。その結果時間発展にともない、対応する記号すなわち疑似アトラクタの種類に関して、複雑な終端状態の構造が発生する様子が観察された (図 4.6B)。特に確率的遷移則 1 が埋め込まれた時終端状態のパターンのエントロピーが、完全に確率的な遷移が達成された時の最高値 $\log_2 3^9 \approx 14.26$ に近い値に収束した。^{*24}これらの結果は、記号力学の予測には、初期値とその時間発展に関して無限精度での観測と計算が要求されることを示唆する。換言すれば、微小な摂動に対してもある一定期間後大きく記号遷移の仕方が変化し、実質的に記号力学を事前に設計者が予測できないことを意味する。

4.4 議論

本章では、Innate Training の手法を駆使して、自発的な記号遷移を高次元カオス力学系上で表現する手法を提案した。また、様々な種類の記号遷移および記号に対応する出力軌道を、3 段階の手法による高次元カオス力学系の部分的な相互作用パラメータの調整で高い操作性で達成されることを示した。ここではまず提案手法のメカニズムに関して議論する。次に提案手法の有効性を、他の関連研究との比較を通して議論する。また提案手法の意義や潜在的な重要性を、複数の観点から議論する。またカオス的遍歴の軌道の設計という観点から、本手法の妥当性を議論する。最後に本研究の目的である自律的な学習システムの設計という観点から、本提案手法がどのように有効に活用されうるかを議論する。

4.4.1 提案手法のメカニズムに対する考察

まず、図 4.3A, B での Innate Training の埋め込み精度の解析は、ESN の次元数が設計できる高次元カオスの長さを制約していることを示唆している。しかしながら、LLE による解析 (図 4.3E) は負の LLE を示す領域が、教師軌道の長さが大きくなるにつれ拡大することを示している。^{*25}これらの結果は、仮に Innate Training において学習がうまくいかない、すなわち教師軌道であるカオス軌道が再現的に生成されなかったとしても、ある複雑な時系列パターンをもつ高次元カオス軌道を各記号遷移に応じて再現的にされることを意味する。そしてその長さは、予め与えられた L_{innate} を超える可能性がある。実際、学習条件 $(M, L_{\text{innate}}) = (3, 1000)$ で調

^{*24} なおエントロピーは 3×3 のグリッドのパターンの頻度から構成される確率分布より計測された。

^{*25} 例えば $L_{\text{innate}} = 5000$ ms の時を参照せよ。

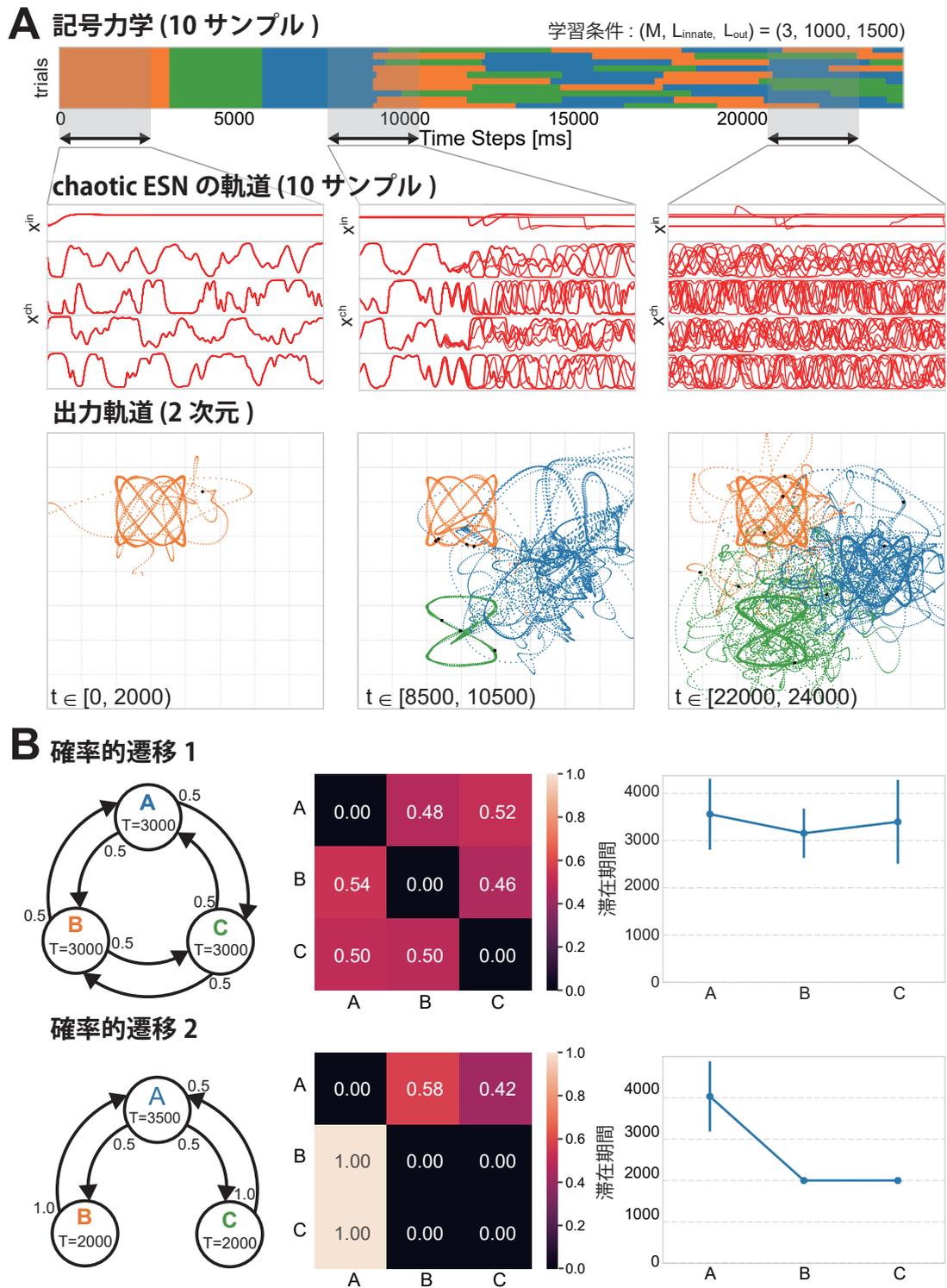


Fig. 4.5 Step 3 のデモンストレーション. (A) Step 3 において確率的遷移を学習した f^{\max} を用いた時の ESN の軌道. (B) 閉ループ f^{\max} の評価. 二つの異なる記号遷移則を目標 (遷移則 1, 2) に関して f^{\max} を学習させた. ESN および Readout は (A) と同じものを使用した. 中央の列の図は得られた確率遷移行列, 右図では記号の切り替え時間が示される.

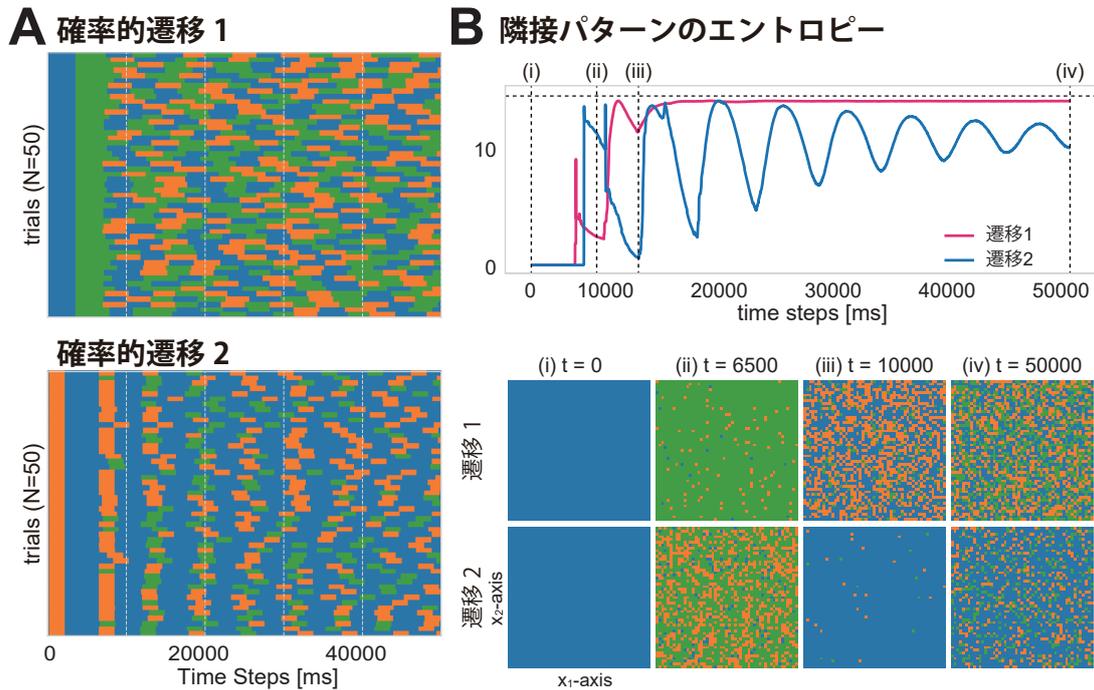


Fig. 4.6 確率的遷移が埋め込まれた高次元カオスの軌道解析。(A) 微小摂動が記号力学に与える影響。図 4.5B で用意された ESN, f^{\max} を評価した。図では初期値の異なる 50 軌道によって生成された記号力学が表示される。(B) ある微小平面の時間発展によって生成される記号力学とその記号パターンのエントロピーの解析。Chaotic ESN を占める次元の内二つの次元 (x_1, x_2) が選択され、微小平面が構成される。そしてその微小平面の時間発展によって生成される記号力学を観察した。また得られた平面の乱雑さを評価するため 3×3 グリッドの模様から、確率分布を構築しそのエントロピーを計算した。なお水平な点線は最大エントロピー $\log_2 3^9 \approx 14.26$ に対応する。

整された同じ ESN を、一連のデモンストレーションで使用したが、 $L_{\text{out}} = 1500 \text{ ms}$ ($> L_{\text{innate}}$) の条件で学習された Readout は安定してリサージュ曲線を出力している (図 4.2, 図 4.4, 図 4.5)。また本実験では、記号遷移に関しても L_{innate} を超える間隔で自律的に切り替えさせた (図 4.4, 図 4.5)。これらは、 L_{innate} より長く再現的に生成される高次元カオスを活用することによって達成される。

その上、この埋め込まれた高次元カオスの局所的に領域の長さが、結果的に設計できる遷移則の範囲を規定すると考えられる。実際、 L_{innate} よりはるかに短い周期で切り替わる記号遷移に関して、今回使用した系はうまく埋め込むことができなかった。また逆に L_{innate} よりはるかに長い周期で切り替わる遷移に関しては、 f_{\max} の学習と挙動が不安定になった。これらの結果は、確率的な記号遷移を安定的に生成させる上で、次の二つの性質が力学系の軌道に要求されていることを示唆する。

1. 確率的な記号遷移を模倣するため、軌道間の違いが切り替え時に十分に拡大され、ばらついている
2. 切り替えのタイミングを制御するため、ある程度軌道の再現性が確保されている

無論これらは相反する性質であり、所望の確率的な記号遷移が安定的に設計されたケースでは、二つの条件が程よく満たされていたと推測される。

4.4.2 提案手法の有効性

まず本提案手法は、複数の過渡的軌道で構成される複雑な遷移パターンの埋め込みに比べて、FORCE 学習や Innate Training と比して高い操作性を有する。実際、複数の遷移を統合し一つの大きなアトラクタとしてみなし、FORCE 学習で学習させたが、多数の遷移で構成される長周期の軌道を埋め込むことはできなかった。^{*26}また誤差逆伝搬法等の力学系全体のパラメータを調整し軌道を構築する手法とは異なり、本提案手法は内在的な高次元カオスを活用し少数の部分的なパラメータのみで達成される。その意味で本手法は誤差逆伝搬法と比してより生物の学習方式に近く、かつその計算量を軽減できる。例えば、運動野に関する神経生理学的研究 [96, 97] によれば、多くの種類の行動が神経系の部分的なパラメータの可塑性で実現される。また提案手法は部分的なパラメータの学習で達成されるため、系全体を学習させる誤差逆伝搬法と比べて計算量的に安価である。これらの性質は即応性とリアルタイム処理が求められる生物規範型の実システムやロボットの設計において重要であり、本提案手法の潜在的な有効性を示唆する。

本提案手法の他のメリットとして、明示的な構造の導入が、高次元カオス力学系において要求されない点が挙げられる。例えば、Namikawa らによって提案された手法 [79, 80, 81] では、階層およびモジュールが制御器内で必要となる。したがって、学習された制御器は特定の行動のみに特化され、他の用途への転用が困難である。また、軌道の生成と記号の切り替えを完全に分離し独立に生成させることも可能かもしれないが、それは低次の軌道が完了するまで記号を保持し続ける外部のメカニズムを要求する。すなわちこのような系では、階層的な構造が暗黙裡に活用されることを意味し、高次元カオス単体では実現できない。対照的に、本提案手法では単一の Chaotic ESN と補助的な記号のみが要求され、かつ様々な用途に活用できる。その上、Innate Training における内部結合の学習は第2章で議論されたように、複数の閉ループの学習と等価であり、多数の Readout で閉ループを組むことによって同等の機能が達成される。したがって、本提案手法は一貫した高次元カオス力学系でさまざまな軌道とそれらの遷移則を設計することを可能にしており、その意味で高次元カオス力学系の応用範囲を大幅に広げる。例えば物理系を Reservoir として活用する PRC (Physical Reservoir Computing) に基づく

^{*26} 付録 B および図 B.3 を参照せよ。

ニューロモーフィックデバイスの設計は本手法の適用先 [51] の候補として有力である。特にスピントロニクスデバイスがカオスダイナミクス [98, 99, 100] を生成することが示されており、PRC の適用候補として脚光を浴びている [101, 102, 103, 104, 105, 106]。このように PRC によってカオス的な物理系を制御する状況で、本提案手法が有効に活用されることが期待される。

本提案手法は先行的に記号を導入している。いくつかの先行研究では、End-To-End 学習^{*27}によってこのような記号的なダイナミクスが自己組織的に表れることが報告されている [83, 107]。これらの手法は確かに先行的な知識を必要としないという意味では有用であるが、このような手法では記号的な構造を積極的に活用できず、学習後に立ち現れるのみである。対照的に、本提案手法は先行的に記号を導入することで、従来手法に比して長周期かつ複雑な機能を設計できる。その操作性において、End-To-End な形式より本提案手法は優れている。

本提案手法は、先行的に M 種類の記号を設定していたが、より高度な設定として記号を生成させるように拡張することも可能である。人間をはじめとする高度な生物知能では、記号の生成と破壊が見られる。特にその発達過程を通して、記号が動的に獲得されるとともに、必要に応じて忘却される。Kuniyoshi ら [108] によって示されているとおり、このような自己組織的な記号力学は、自動ラベル付け機構を外的に追加することで実現される。また、[109, 110, 111] で導入されているような、時系列から離散的な状態を抽出するための教師なしアルゴリズムを系に組み込むことで、自発的な記号の生成が可能となる。^{*28}

4.4.3 提案手法の重要性

まず提案手法の持つ高い操作性は、脳の情報処理の根底にあるメカニズムを特定の観点から理解するのに役立つだろう。第 1 章で議論されたように、動物の神経系は膨大な数の非線形素子とそれらの相互作用で成立し、結果的に非常に複雑なかつ自発的な活動を生み出す。特にマクロ的な階層（細胞の活動電位など）からマクロ的な測定レベル（脳波など）までの幅広いスケールにおいて、カオス的な挙動が現れることが報告されている [30, 31]。本章では高次元カオスの潜在的な表現能力が明らかになったが、これは高次元のカオス的な脳活動が、それぞれのスケールで階層的な機能性を実現できることを示唆する。このように本提案手法および構築されたモデルは、動物の脳が生み出す高次元カオスが与える情報処理への寄与を理解するための手掛かりになるだろう。

本提案手法は、単にカオス軌道を利用した従来手法とは本質的に異なる利点を有する。本提案手法によって構成された高次元カオス力学系は、局所的な安定性と大域的なカオス性が共

^{*27} 入出力関係だけを提示し、途中の処理全てを学習させる形式。ここではニューラルネットワークのパラメータの学習を意味する。

^{*28} なおこの発展形に関しては第 6 章と第 8 章においてより詳細に議論される。

存する興味深い性質を有する。この性質は、適応性と自己同一性の両方が求められる認知モデルを設計する際に特に有効である。例えば、ロボットの安定性を維持しつつ、マクロ的な予測不可能性を利用して自発的な動作プリミティブの切り替えを実装することができる。さらに本提案手法は、確率分布つまり大局的な行動の傾向の設計に役立てられることが期待される(図4.5)。また、動物の認知機能(記憶の想起や連想など)は、定型的な活動の遷移現象によって実現されていることがいくつかの研究で指摘されており[29, 48], 抽象的なレベルで共通点があるといえる。このように認知モデルを高次元の力学系に実装するために本提案手法は活用されうるだろう。

最後に大自由度力学系の観点から本手法で得られた現象の意義に関して議論する。本論文では、階層的なダイナミクスが単一の力学系から生成されているため、ミクロな変化がマクロな確率遷移に影響しうることを示した。具体的には図4.6で明らかにされたように、 $\mathbf{x}(t)$ の初期値の微小な差分が系のカオス性によって拡大され、記号遷移の仕方が大幅に変化する。このようなミクロな変化のマクロなダイナミクスへの影響は、高次の記号遷移が低次元層とは独立したメカニズム^{*29}で制御されるシステムでは原理的に生じえない。また力学系ではしばしば平均場理論を用いて系の特性が解析されるが、平均場近似による力学系の解析ではこのようなミクロな変化がマクロなダイナミクスに与える影響は無視されがちである。このように本手法によって構成された自発的に記号を切り替えるシステムは、決定論な力学系におけるミクロとマクロなダイナミクスの相互作用に光を当てるもので、有意義である。

4.4.4 「カオスの遍歴の設計手法」としての提案手法の妥当性

冒頭で議論されたとおり、本提案手法により実現された自発的な記号遷移は、一種のカオスの遍歴と解釈できる。そこで本手法を用いて設計したカオス軌道のカオスの遍歴としての妥当性について、過去のカオスの遍歴の研究と比較して議論する。

まず本章ではESNのカオス性を利用することで、目標の遷移則を模倣できることを示した。しかし、いくつかのカオスの遍歴の研究では、カオスの遍歴の遷移が履歴依存的であることが指摘されている。例えばKanekoとTsuda[112]は、カオスの遍歴の遷移の仕方がランダムとは区別されることを指摘している。ItohとKimoto[113]は遷移の仕方において偏りが生じることを報告している。実際、今回構成された高次元カオス力学系でも、切り替え前の記号に応じて出力軌道に偏りを生じることが確認された。^{*30}その意味で本提案手法によって埋め込まれる記号力学は履歴依存的であるといえる。

さらにそのような偏りが存在するにもかかわらず、本提案手法によって記号の遷移のランダム性をうまく表現されていることは強調されなければならない。付録Bにおいて示される解

^{*29} 例えば独立な確率変数等。

^{*30} 詳細は付録Bを参照せよ。

析結果は、遷移則 1, 遷移則 2 いずれの場合においても遷移の確率が前の記号によってそこまで変化しないことを示している。これは、マクロ的な記号の履歴を参照するだけでは、次の遷移の予測は依然として困難であることを示唆している。このように高次元カオスの軌道の上では偏りが観察されるものの、マクロ的な記号遷移は依然としてランダムに見える。したがって、本提案方法は系のカオス性を利用することにより、高次元非線形力学系上で遷移確率とそのランダム性を再現しているといえる。

本提案手法で、有限個の擬似アトラクタで特徴づけられるランダムな遷移 [80, 114] は、記号を対応させることで実現可能である。一方でカオスの遍歴の軌道の中には、本提案手法でも設計が困難なものがある。例えば Tsuda ら [115] はロングテールな分布で特徴づけられる遷移の仕方をするカオスの遍歴を報告した。また擬似アトラクタを無数に生成し、無限状態機械^{*31}でのみ特徴づけられるようなカオスの遍歴の例も報告されている [88]。このようなカオスの遍歴を表現するには、無限個の記号と対応する無限個の高次元カオス軌道が用意される必要があり、本提案手法ではその実現は困難である。

最後に遷移状態からの回復速度の観点から本手法を論じる。Ahmadi と Tani[116] は PV-RNN (Predictive-coding-inspired Variational RNN) と呼ばれる確率モデルを用いて、プリミティブ間の確率的な遷移を模倣させた。そこではアトラクタの安定性は、目的関数の中で用いられるベイズ推定の変分下限の複雑さの係数によって決定される。したがって、この例では目的関数を定式化しそこに現れる係数を用いることで、回復速度を調整できる。この点に関して本提案手法における系の安定性は、ESN のスペクトル半径および Input ESN と Chaotic ESN の間の接続強度によって決定されると考えられる。第 2 章で議論された通り、ESN のスペクトル半径が 1.0 を超えると、非カオス領域からカオス領域への遷移が発生する。また ESN のカオス性はスペクトル半径の増加によって強化され、スペクトル半径が大きくなるほど回復速度が遅くなると推測される。さらに共通入力同期により、Chaotic ESN への投射強度が大きいほど、Chaotic ESN の内部状態が同期しやすくなる。したがって、遷移状態からの回復速度は、スペクトル半径および Input ESN と Chaotic ESN の間の投射強度の調整によって調整できる。まとめると、ESN のスペクトル半径と接続強度によって、遷移状態からの回復速度を今回の系でも説明し、かつある程度制御することができる。

4.4.5 自律的な学習システムの構成における提案手法の寄与

本提案手法を用いることにより、これまで外的に生成されてきた入力を選択、すなわち記号の切り替えを ESN 自身によって制御させる構成が可能になった。特に確率的な記号遷移則を用いることで、自発的な記号の切り替えを行う高次元カオス力学系が構成される。この高次元

*31 すなわち記号が有限ではない。

カオス力学系を内包させることで、ある遷移則で入力切り替わる **Innate Training** を、さらに大きな高次元非線形力学系のダイナミクスとして表現することができる。このように本提案手法を駆使することで、現在は設計者によって制御されている入力の切り替えを、学習システム自身が自発的に行うような系の構成が可能となる。一方、依然として記号の種類や遷移則は、設計者によって先行的に指定される必要がある。これらの非自律的な要素は、記号をいわば動的に生成させることで解消されると期待される。この点に関しては、今後の研究の展望として第8章において議論される。

第 5 章

Neuralized All-In-One Reservoir の提案および高次元力学系への学習過程の埋め込み

[特許出願準備中のため非公表]

第 6 章

Developmental Innate Training の提案およびその動的特性

本章では、Innate Training では外的に制御・生成される教師軌道が、ニューラルネットワークのダイナミクスによって自律的に構成されるように拡張された DIT (Developmental Innate Training) を提案する。そしてデモンストレーションならびに数値解析を通して DIT の特性を調査する。最後に DIT の特性に関してその有効性を議論する。

6.1 問題設定

Innate Training はある入力に対する Chaotic ESN の応答を再現的にする手法である。この際、初期ネットワークによって生成された高次元カオス軌道は教師軌道として繰り返し学習において活用される。第 2 章において議論された通り、この Innate Training の事前学習の設定を実現するには、初期ネットワークと教師軌道の初期値を保持し、教師軌道を生成し続ける機構が学習システムの外側に必要となる。このような構成は自己充足的でなく要件 3 に抵触する。またこの教師軌道の初期値は通常設計者によって先行的に決定される。通常特に指定なくランダムにこの初期値が策定されるが、そこには依然として設計者による恣意性が存在する。要件 2 の自己決定・構成能を向上させるためにも、このような初期値の設計者による選定が不必要になる構成が望ましい。さらに、Innate Training では教師軌道の長さが先行的に与えられる。言い換えればその範囲の外では学習は行われず、結果的に学習の停止と進行が設計者によって制御されている。要件 1 が求める自発的な学習の達成のため、設計者に依存せず常に学習が進行し続ける構成が望ましい。

本章では、Innate Training の教師軌道が学習システム内部のダイナミクスによって動的に生

成される DIT とよばれる構成を提案する。^{*1}具体的には図 6.1A に示されるような教師軌道のやり取りを行うサブモジュール (Chaotic ESN + RLS アルゴリズム) のネットワークを考え, Chaotic ESN のダイナミクスが, 別のサブモジュールのあるノードの教師軌道として動的に投射される構成を DIT として採用する. 一方でこの DIT におけるサブモジュールのネットワークの形態は無数に考えられる. そこで議論を簡単にするため, 特に典型的な DIT のレパートリとして, 2 サブモジュール間の双方向および片方向モチーフを想定し (それぞれ以下双方向 DIT, 片方向 DIT と呼ぶ) その特性を解析する. このような DIT の構成では, 教師軌道の生成が高次元非線形力学系内で動的に行われるため, 自己充足的である. また設計者による教師軌道の初期値の設定が不必要になる. さらに設計者によって外的に止められない限りは学習が恒常的に進行し続ける. このようにこの DIT の構成は, 少なくとも冒頭に挙げられた要素が解消された構成であるといえる.

さてここで問題となるのが DIT の学習特性, すなわち Innate Training 同様この DIT が入力に対して局所的に再現的なカオス軌道が生成されるか否かである. そこで本章ではまずデモンストレーションによって双方向, 片方向 DIT の学習特性を調査する. そして双方向 DIT では, 系全体が非カオス的すなわち固定点やリミットサイクルに収束し期待される学習の効果が得られない一方, 片方向 DIT では非カオス的な状態とカオス的な状態を定常的に往来し, 恒常的に学習が進行することを示す. そしてその片方向 DIT において, Innate Training と同様局所的に再現的なカオス軌道が入力に応じて生成されることを確認する.

さらにこの片方向 DIT の定常的な特性を複数の指標を用いて評価する. まず学習される系の力学系的特性の変化を明らかにするため, MLE によってカオス性の推移を計測する. また Innate Training による学習効果の変化を評価するため, ESP Length と呼ばれる指標を導入しその遷移を観測する. これらの解析を基に, 本章では片方向 DIT が様々な側面でスケールフリーな特性を有することを示す. 最後にこの片方向 DIT が示すスケールフリー性の機能性を複数の観点から論じる.

6.2 提案手法

6.2.1 双方向・片方向モチーフ

DIT は Innate Training の教師軌道を Chaotic ESN 内のノードにより動的に生成させることによって実現される. これは式 (2.17) 内の教師軌道 $\mathbf{x}^{\text{target}}(t)$ を Chaotic ESN の内部ダイナミクス $\mathbf{x}(t)$ で置換する操作に等しい. 一方でこの置換の方式は無数のパターンが存在し, その全てを網羅的に調査することは困難である. そこで簡単のため, まず図 6.1A の模式図に示され

^{*1} “Developmental” は教師軌道が時間発展することを強調している.

るように、同じノード数とネットワークトポロジーを共有する複数のサブモジュール^{*2}の間で教師軌道をやり取りすることを考える。さらにネットワークモジュールの数を限定し、その中の基本単位である双方向・片方向の二種類のモチーフに着目する (図 6.1B)。本章ではこの二つのモチーフにおける DIT の動的特性を解析する。

6.2.2 詳細な実験設定

実験では、同じ結合行列を共有する二つの Leaky ESN ($N = 800, \tau = 10.0, g = 1.6$) をサブモジュールとする系を使用する。Leaky ESN の内部結合 J の各要素は結合密度 p で標準正規分布よりサンプルされ、 $\rho(J) = 1.0$ となるように定数倍される。各サブモジュールの内部結合は共有されるが、その初期状態は異なる値に設定される。この初期値の違いとネットワークのカオス性 (初期値鋭敏性) により、時間発展にともない内部結合が同期せずに異なる値を取ることが想定される。また系に与えられる入力として 5000 ステップ毎に 50 ステップにわたって定数ベクトル \mathbf{u} (一様分布 $\mathcal{U}([-1, 1])$ よりサンプル) が投射される周期的矩形波を採用した。この入力は二つのネットワークに同様に投射される。そして矩形波の間の 5000 ステップの期間を 1 エポックと定義し、ダイナミクスの長期間にわたるサンプリングを通して系全体の定常的な特性を調査する。^{*3}

双方向モチーフの設定では 800 次元の内、前半 K ノードの結合パラメータが両方の ESN で学習される。この時、互いの各次元のダイナミクスが、互いの教師軌道として投射され学習が進行する (図 6.1B 上)。片方向 DIT では対照的に一方の ESN (Parent System) の内部結合は固定化され、そのダイナミクスがもう片方の ESN (Child System) の教師軌道として投射される (図 6.1B 下)。また DIT では RLS アルゴリズムが Chaotic ESN 内の各ノードで別々に動作しているが、RLS アルゴリズム自体の ESP を担保するため忘却率 μ が 1 未満に設定される。特に実験では $\mu = 0.9999$ を採用した。

6.3 結果

まず、双方向・片方向モチーフによる DIT の典型的な学習特性を調査する。図 6.2 は双方向・片方向モチーフの両者について、 $(K, \mu, p) = (200, 0.9999, 0.05)$ の条件で 512 エポック^{*4}の間学習させた時のデモンストレーションである。上図では、各エポック後の内部結合で固定した際に計測された Leaky ESN の MLE の遷移が示されている。^{*5}まず双方向モチーフでは一

^{*2} いずれも Chaotic ESN.

^{*3} なおこの DIT の設定では内部で Innate Training が常に進行し続ける。すなわち結合行列が常に変化し続ける。

^{*4} すなわち $512 \times 5000 = 2560000$ ステップ。

^{*5} なおこの MLE は各エポックにおける一時的なスナップショットであり、DIT 全体のそれとは異なる。あくまで各エポックで、各サブモジュールである ESN がどのような特性を有しているのかが評価されている。

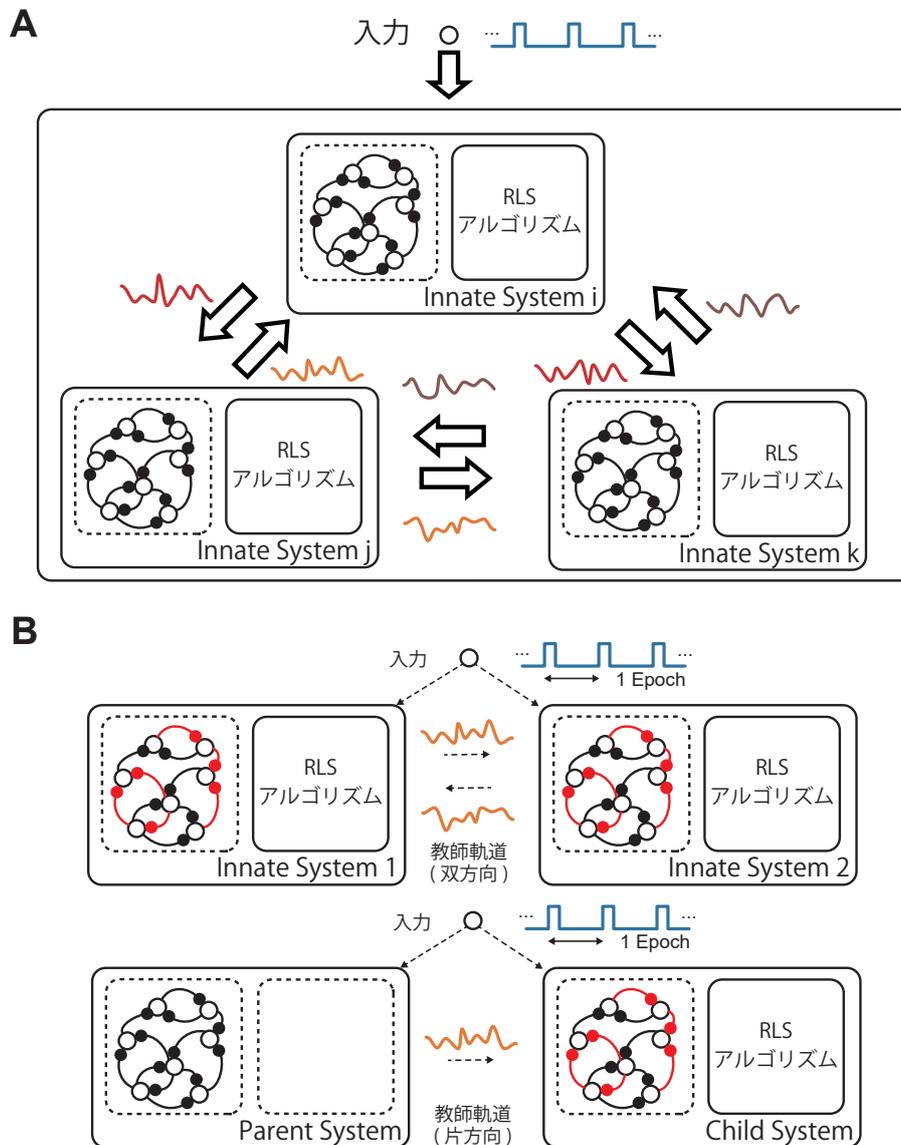


Fig. 6.1 DIT (Developmental Innate Training) の模式図. (A) DIT の定義. DIT は Chaotic ESN はお互いに軌道を渡しあうサブモジュール (ESN) のネットワークとして解釈できる. 本章では特に, 各サブモジュールの内部結合が Innate Training によって時間発展し, その教師軌道が他モジュールの軌道より動的に生成される DIT を考える. (B) 2 モジュール DIT におけるネットワークモチーフ. ここで二つのモジュールに着目した場合, 右図に示されるように, 双方向・片方向の二種類のモチーフが考えられる. 実験では入力として一定期間間隔投射される矩形波を採用し, 矩形波が投射される間隔を 1 エポックと定義する.

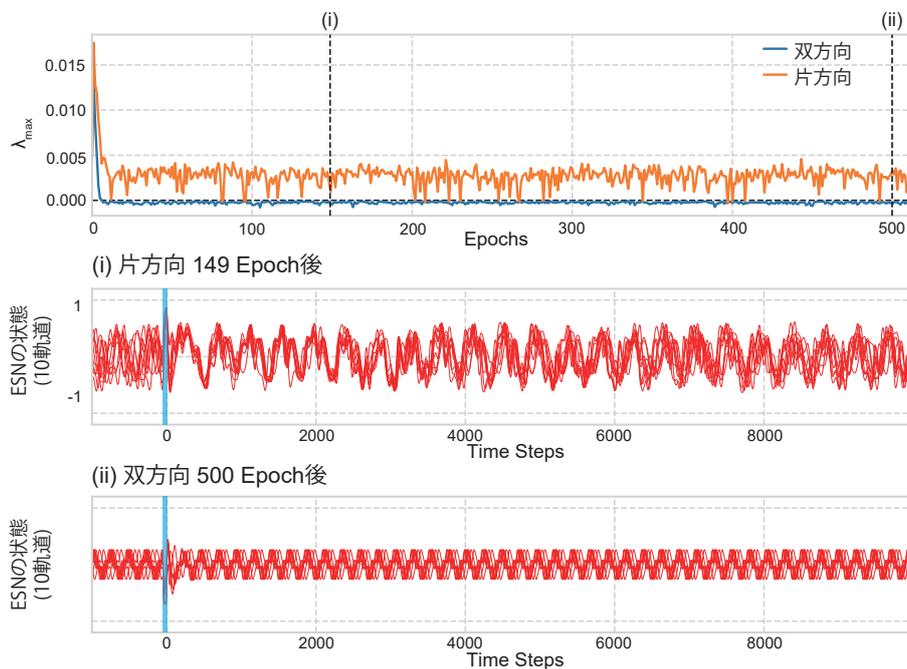


Fig. 6.2 DITのデモンストレーション. 双方向・片方向モチーフの両者に対して, $(K, \mu, p) = (200, 0.9999, 0.05)$ の条件で 512 エポック学習させた. 上段では各エポック後の結合行列を用いた時の, MLE (λ_{\max}) の遷移を表す. 下段は得られた結合行列を用い矩形波入力を $t \in [-50, 0]$ (水色) で与えた際の ESN の軌道を表す (初期値の異なる 10 軌道がそれぞれ重ね描かれている).

度 0 に到達した後, ほとんど MLE が変化しないことが分かる. また下図に入力に対する応答から分かる通り, 得られた結合行列による Leaky ESN は非カオス的でリミットサイクルを生成している. これは両方のネットワークが非カオス状態に遷移しほとんど同期し学習がほぼ停止していることを意味している. 実際, 実験で観察された限りでは, この双方向 DIT の設定ではリミットサイクルや固定点に収束するケースのみが観察された. 対照的に片方向 DIT では, はじめ大きく値が下がるものの MLE の遷移が正の領域で定常的に安定した. また時折負の領域への遷移が見られるものの, 学習の進行にともない正の領域へ回復した. この恒常的なカオス性は, 結合行列が固定化されたネットワーク (図中の Parent System) においてカオス性が担保されることに起因していると考えられる. 加えて, 得られた結合行列による系の入力応答について, Innate Training で期待されるような局所的に安定的なカオス軌道が生成された (図 6.2). ここで, この片方向 DIT では片方向的に与えられる教師軌道がエポック毎に動的に変化することに注意されたい. すなわち, 通常の Innate Training とは異なり, RLS アルゴリズムにおいて常に誤差が発生しうる状況であると考えられる. これらの結果は, 常に誤差が生じ RLS アルゴリズムによる学習が完全には達成されない片方向 DIT の構成でも, 再現的なカオス軌道の生成という観点では学習の効果が生じることを意味する.

次にこのデモンストレーションを踏まえ、以下片方向 DIT に関してより詳細に調査する。まず、図 6.2 同様、得られた結合行列を用いて各エポックにおける MLE を計測し、系のカオス性の変化の特性を評価する。より確実に特性を評価するため、 $(K, \mu, p) = (200, 0.9999, 0.05)$ の条件で生成された 5 個のネットワークそれぞれに対し 8192 エポック^{*6}学習を行い各エポック後の結合行列に関して MLE を計測した。図 6.3A はその 1 サンプルに関する遷移である。図 6.2 で観察されたように定常的にはカオス状態で、たまに非カオス状態 ($\lambda_{\max} < 0$) への遷移とカオス状態への回復が生じた。実際この MLE に関して分布を計測したところ正の領域に偏った分布が得られたが、これは片方向 DIT が平均的にはカオスであることを意味する。また MLE の遷移に関して、非カオス状態への移行が間欠的に生じているように見える。この非カオス状態への移行イベントが生じる間隔に関して分布を計測したところ、両対数グラフにおいて直線にフィッティングされるような分布が得られた (図 6.3C)。これは得られた非カオス状態への移行イベント間隔に関する分布がスケールフリー性を有することを示唆する。^{*7}

MLE によるカオス性の評価に加え、DIT の学習特性の遷移を、ESP Length と呼ばれる指標を導入して評価する。ESP Length は入力を与えられた後の安定的な軌道の期間によって定義づけられ、Innate Training の学習の効果の度合いを計測する指標である^{*8}。具体的には、異なる初期値の 10 軌道間の分散が閾値 (10^1) 以下で維持される期間によって計算される。例えば図 6.4A に示されるように、初期結合行列に比べて DIT 途中の ESP Length は大きな値をとる。図 6.4B は各エポックに対する ESP Length の遷移を表したものである。上端に振り切れている点は、系が非カオス的になり閾値以上に分散が回復しなかったケースで、観測上限である 10000 ステップが ESP Length の値として与えられている。まずこの ESP Length の大きさに関して分布を構築したところ、先ほどの非カオス的間隔の解析と同様、両対数グラフ上で直線上にフィッティングされるような冪的な分布が得られた (図 6.4C)。^{*9}また ESP Length が 500 以上になるエポックを Innate Training の学習の効果が反映されている安定状態と定義し、その間隔を同様に計測したところ、こちらも冪的な分布が得られた (図 6.4D)。このように Innate Training による学習特性においても、この片方向 DIT の設定ではスケールフリー性が様々な側面で観測された。

^{*6} すなわち $8192 \times 5000 = 40960000$ ステップ。

^{*7} なお他の (K, μ, p) のパラメータに対しても類似のスケールフリーな分布が観測された。詳細は付録 D を参照せよ。

^{*8} すなわち ESP Length が大きいほど Innate Training の効果があると考えられる。

^{*9} ESP Length が小さい分布に関して直線から離れているが、これは矩形波入力による共通入力同期によってある一定の ESP Length が確保されていることに起因していると考えられる。すなわちある一定以下の ESP Length が生じず、結果的にその箇所に関して直線を下回っている。

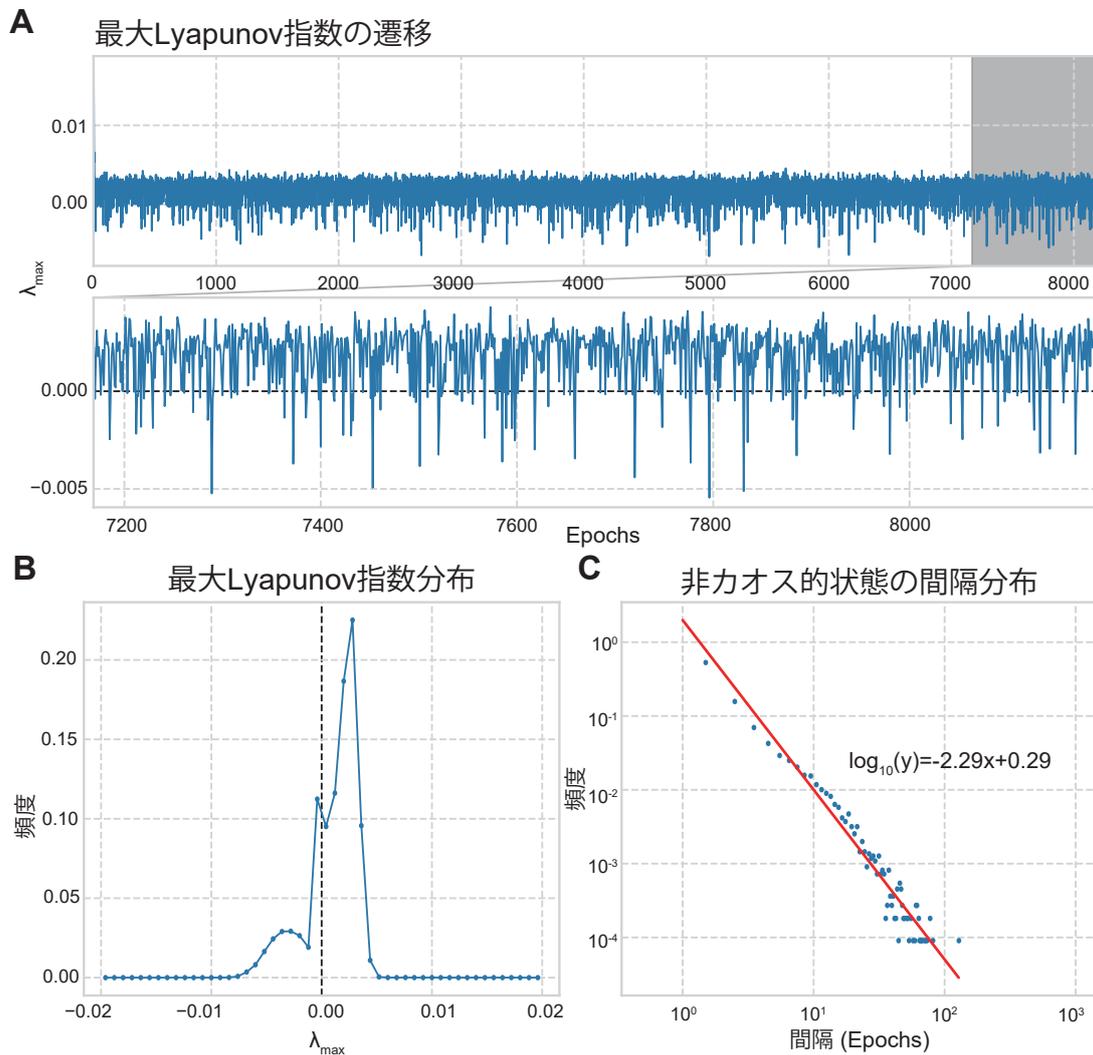


Fig. 6.3 MLE (最大 Lyapunov 指数) による解析. (A) MLE の遷移の一例. $2^{13} = 8192$ エポック学習させ, 各エポックで得られた結合行列に対してそれぞれ MLE を計測した. (B) MLE の分布. 5 個の初期結合行列を用意し, それぞれ 8192 エポック学習させ MLE を計測した. (C) Non-Chaotic 間隔分布. ESN が Non-Chaotic ($\lambda_{\max} < 0$) になるイベント間の間隔に関する分布が示される. 赤線は両対数グラフに関して線形にフィッティングされた直線を示す.

6.4 議論

まず本章で検証された両方向・片方向のモチーフの効果に関して議論する. 実験では両方向 DIT が非カオス的状態への遷移を引き起こし Innate Training で期待される学習の効果が得られなくなるのに対し, 片方向 DIT では安定してカオス状態が維持され, 恒常的に学習が進行した. これは DIT, より一般に学習システムにおける順方向的なトポロジーの重要性を示唆し

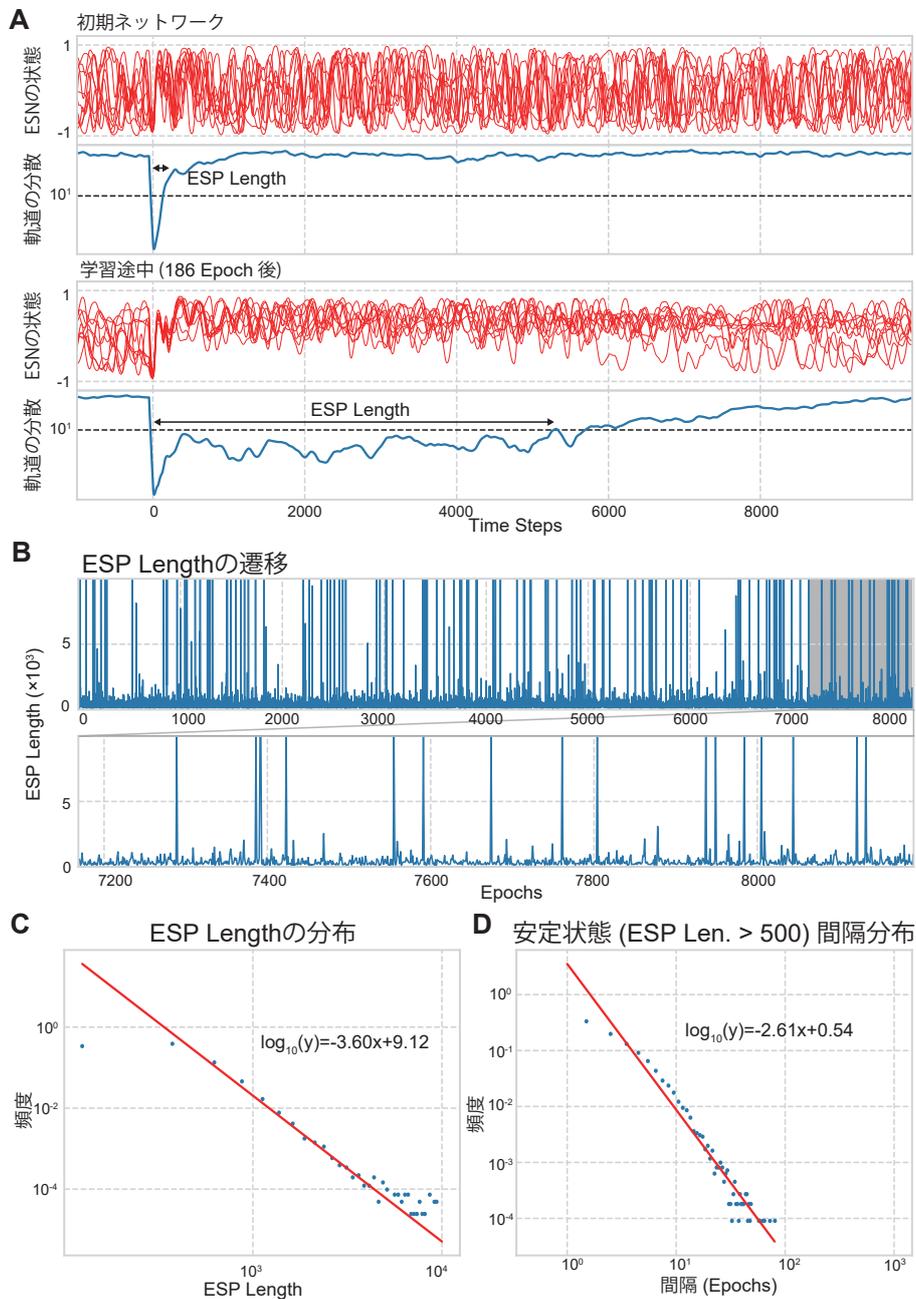


Fig. 6.4 ESP Length による解析. (A) ESP Length の定義. ESP Length は 10 個の異なる初期状態から時間発展する軌道に関して、矩形波入力を投射されたのちの軌道の分散の復帰時間により定義される. 特に実験では軌道の分散が 10^1 を超えるまでの時間を ESP Length として計測した. (B) ESP Length の遷移の様子. 各エポックの結合行列に関して ESP Length が計測された. 固定点に収束する等の要因で軌道の分散が閾値以下を取り続けた時、ESP Length は無限大となる (図の上端に達している点に対応する). (C) ESP Length の分布. 赤線は両対数グラフに関して線形にフィッティングされた直線を示す. (D) 安定状態間隔の分布. ESP Length が 500 以上になる点を安定状態と定義し、その間隔に関する分布が表示される. 赤線は両対数グラフに関して線形にフィッティングされた直線を示す.

ている。すなわち、学習システムが常に自発的かつ恒常的に学習を達成するために、カオス的な要素が安定的に内包され、かつそれが片方向的に投射される構成が肝要であることを示唆する。またこの間に挟まる Chaotic ESN は、入力に応じて局所的に応答を生成し別ネットワークに投射されるフィルタとして機能しており、Input Reservoir の一種とみなすことができる。^{*10} この特性は第7章の「統合学習システム」の構成においても活用される。

次に片方向 DIT において観察されたスケールフリー性について、その機能性を議論する。今回非カオス状態への移行 ($\lambda_{\max} < 0$) や、Innate Training の効果の発現 (ESP Length > 500) といった一種の秩序状態への形成に関してスケールフリー性が観察された。これは今回の DIT が、一種の自己組織化臨界現象 (Self-Organized Criticality, SOC) [117] を引き起こすことを示唆する。SOC は自発的に構造を形成する自己組織化系において観察される現象である。古典的には砂山崩しモデル [117] における雪崩現象が SOC を生じる数理モデルとして導入されたが、その後自然界、特に開放系において多く観察されることが報告されている。^{*11} 今回の片方向 DIT でも、教師軌道が時々刻々と時間発展しその結果、学習の達成と崩壊を繰り返す一種の臨界現象が起きていると解釈されうるが、その検証にはより系統的な解析が必要である。

今回スケールフリー性が、ある単一モチーフのネットワークの時間的な分布の中で観測された。一方これは図 6.1A で示される通りこのモチーフはあくまで非常に限定化された DIT のモデルで、より一般的には脳のように多数の学習するサブモジュールが相互に結合した系を考えることができる。そのような巨大な系では、今回着目した双方向・片方向モチーフが数多く空間的に分布していると考えられる。したがって、今回得られた秩序状態に関するスケールフリーな特性が、巨大な DIT 系では空間的な特性としても立ち現れることが期待される。言い換えれば、任意の時刻で入力に対して安定的にカオス軌道を生成するようなモジュールが、冪的な分布で空間的に発見されることを意味する。これはそのような巨大な DIT の系が、様々な時間スケールの時系列をどの時刻においても生成しうることを示唆する。

今回の実験で、間欠的な安定状態への遷移が DIT において観察された。これはカオスの遍歴状のダイナミクスが、DIT において観察されうることを示唆する。実際安定状態では、系は再現的なカオス軌道を生成できるので、DIT 全体ではその軌道を擬似アトラクタとするカオスの遍歴状のダイナミクスが生じているとも解釈できる。今後の研究で、他のハイパーパラメータ (K, μ, p) においても、同様にカオスの遍歴状のダイナミクスが生じるか検証される必要がある。

今回の DIT では、教師軌道がネットワークによって動的に生成されかつ系のカオス性によって時々刻々と変化する。したがって通常の Innate Training とは異なり恒常的に誤差が大きくなり、その意味では学習はうまくいっていないといえる。一方でそれにかかわらず今回の系では

^{*10} ただし第3章と異なりこの Input Reservoir はカオス的である。

^{*11} 地震、森林火災の広がり、脳・神経活動 [118] 等。

局所的に Innate Training 同様の学習の効果が表れ、かつ入力に対して安定的にカオス軌道を生成されるイベントが冪的な分布で発生することを示した。この事実は、誤差そのものが最小化されなくても、Innate Training および DIT の時間発展そのものが学習の効果をもたらさうることを示唆している。^{*12}したがって今後 DIT をさらに発展させ、誤差やそれに基づく目的関数の最適化にとらわれない、新たな学習システムの形態が実現されることが期待される。そのような形態は学習システムや学習機能に対する新しい視点を提供するだろう。

^{*12} ここでの学習は無論「誤差の最小化」ではなく、「入力に応じた再現的なカオス的応答の埋め込み」を意味する。

第 7 章

統合学習システム—自律的に学習する 高次元カオス力学系の構成

本章では、ここまでの章で提案された Input Reservoir, Neuralizing, DIT (Developmental Innate Training) の手法を駆使・統合し, Innate Training そのものが埋め込まれたニューラルネットワークの設計手法が提案される. その手法に基づき構成された「統合学習システム」において, Innate Training 同様, 入力に応じて局所的に安定的なカオス軌道を生成するように系が学習されることを示す. また AIOR (All-In-One Reservoir) の発想を活用し, 機能性を保持したまま統合学習システムの縮退した表現が可能であることを示す. 最後に第 1 章で定義された要件と照らし合わせ, 実現された統合学習システムの自律的な学習システムとしての妥当性を議論する.

7.1 問題設定

第 6 章で導入された DIT では, 教師軌道が Chaotic ESN のダイナミクスによって置換されることにより, 要件 1, 要件 2, および要件 3 が克服される形で Innate Training が拡張された. また DIT では入力を Parent ESN に投射して教師軌道が生成されるため, この Parent ESN がフィルタとして一種の Input Reservoir とみなされる. 残る DIT の非自律的な要素は, RLS アルゴリズムと Chaotic ESN の分離構成で, これは AIOR と Neuralizing を駆使して解消される. すなわち RLS アルゴリズムがニューラルネットワークとして表現されることで自己充足性が回復され, 要件が完全に満足された自律的に学習システムの設計が達成される.

本章では片方向 DIT を基盤に据え, その中の RLS アルゴリズムおよび内部結合が Neuralizing によってニューラルネットワークに置換された Neuralized Innate System (NIS) を構築する. そして複数の NIS を組み合わせることにより, 片方向 DIT の全ての要素がニューラルネットワークとして表現された Neuralized DIT が完成する. この Neuralized DIT は自律的な

学習機能が埋め込まれたニューラルネットワークで、本章ではこれを「統合学習システム」と呼称する。まずこの統合学習システムが入力に応じて **Innate Training** 同様に、時間発展にともない局所的に安定的なカオス軌道が生成されることを示す。さらに **AIOR** の着想を応用し、複数個存在する **NIS** の一部を活用し、線形近似により残りの **NIS** を置換する。実験ではこのように部分的に線形に近似された系でも、第6章で観察されたようなカオス的狀態と非カオス的を行き来する臨界現象が生じる事を確認する。

7.2 提案手法

提案する統合学習システムの全体像は図 7.1A に示される。 N 次元の **Chaotic ESN** の内、 K 次元のノードが学習される片方向 **DIT** の系では、各ノードに対応して K 個の **RLS** アルゴリズムが並列で動作していると解釈できる。本章では、これを **Neuralizing** によって4層のニューラルネットワークとして表現された **NIS** (図 7.1B) を用意する。この **NIS** の構築の過程は第5章の **Neuralized RLS/Readout** とほぼ等価であるが、**ESN** の内部結合における内積計算用のモジュールが最終層に加えられている。^{*1}結果的にこの統合システムは教師軌道を生成する **Parent ESN**、ネットワークの固定結合ノードを実現する、**Child ESN**、そして K 個の可変結合ノードならびに **RLS** アルゴリズムを内包する **NIS** の三つの部分に分割される。そしてこれら全ての要素は **tanh** の非線形性のみによって実現されており、一貫している。

7.3 結果

実験では $N = 20, g = 2.0$ の離散 **ESN** を **Parent ESN** と **Child ESN** の構成のために用意した。そして **Child ESN** の 20 ノードの内、 $K = 5$ ノードを選択・分離、その可変結合および **RLS** アルゴリズム ($\mu = 0.9999, \alpha = 1.0$) を実現する K 個の **NIS** を用意した。まずこのようにして構築された統合学習システムの妥当性を、デモンストレーションを通して示す。具体的には **Innate Training** 同様入力に対して局所的に安定的なカオス軌道を生成できるかどうか検証する。そして入力として第6章同様、矩形波の繰り返しで表現される周期的な入力を採用した (500 ステップ周期, 50 ステップの定数入力)。^{*2}図 7.2A は学習中の統合システムの内部ダイナミクスを表示している。各凡例は **NIS** の番号に対応し、 $K = 5$ 個全ての **NIS** のダイナミクスが表示されている。図から分かるように、あるニューラルネットワークのダイナミクス

^{*1} この構成は結合行列に関する高階な設定 (内積計算がニューラルネットワークの「外で」別途用意される) を省くために導入される。結果的に **Child ESN** は固定結合部と可変結合部に分割され、可変結合部は **NIS** によって高階な設定を無くした状況で実現される。

^{*2} 無論この周期的入力も究極的には第4章を用いて自律的に生成させる構成を取ることも可能である。しかし本実験では得られた統合学習システムの学習性能を評価するため、あえて入力の一例としてこのような信号を採用した。

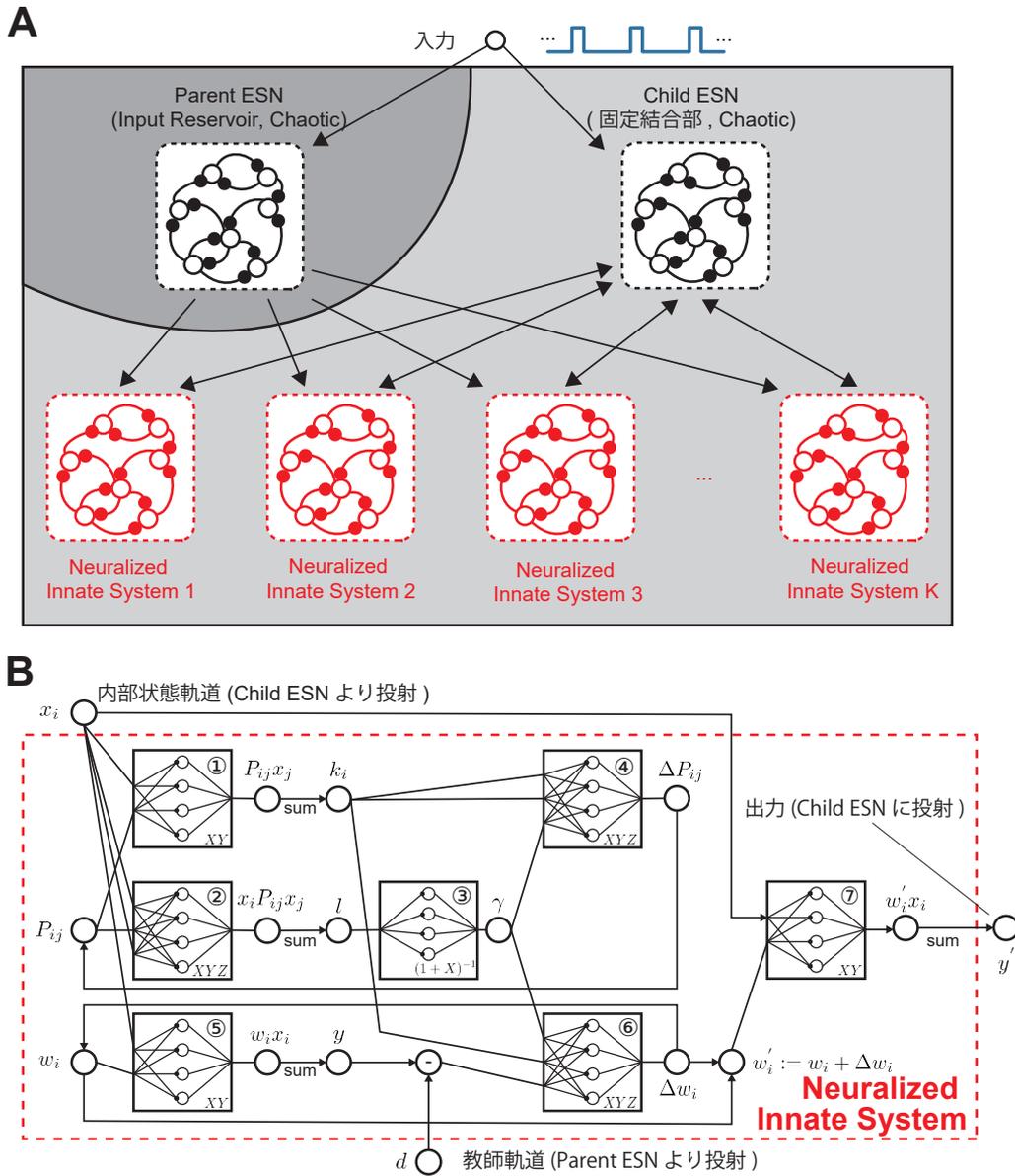


Fig. 7.1 統合学習システムの模式図。(A) 統合学習システムの全体像。DIT を主体に全ての要素が Neuralizing を駆使してニューラルネットワークとして表現されている。具体的には Innate Training 中の RLS アルゴリズムおよび可変結合が NIS (Neuralized Innate System) として表現されている。また入力が Parent ESN に投射され、そこで生じたダイナミクスが他の要素に投射される構成をなす。したがってこの Parent ESN は Input Reservoir の一種と解釈できる。(B) NIS の概要図。各々の非線形演算は第 5 章同様 ELM によって表現される。Child ESN より内部状態軌道、Parent ESN より教師軌道を入力として受け取る。

の結果，元の Child ESN の内部結合に対応する $W(t)$ が時々刻々と変化している様子が読み取れる．また $t = 0$ と $t = 10500$ に得られた $W(t)$ の値を用い，離散 ESN の内部結合に挿入し矩形波入力に対する応答を評価したところ，より長い期間安定的に複雑な軌道が生成された (図 7.2B).^{*3}これはこの統合学習システムが，そのニューラルネットワークのダイナミクスを通して Innate Training と同様入力に対する応答を再現的にできる事を意味し，得られた統合学習システムの妥当性を支持する．

さらにニューラルネットワークとして表現されることの利点を議論するため，統合学習システムの内部状態を計算資源，すなわち Reservoir として活用して，状態の次元削減に活用できることを示す．すなわち第5章で導入された AIOR の着想を応用し，統合学習システムのより縮退された形態を模索する．具体的には実験では NIS 1, 2, 3 の内部状態を用いて，NIS 4, 5 内の $P(t)$ のダイナミクスを近似させるような閉ループを構築する (図 7.3A).^{*4}Neuralizing によって生じた高次元の隠れノードのダイナミクスの効果を検証するため，第5章の実験同様，NIS 中の $P(t)$ のみを使用する条件と全状態を使用する条件を比較した (それぞれ「 P のみ」，「全変数」と表記)．図 7.3B は実際の各 NIS のダイナミクスの様子である.^{*5}まず学習データセットに対する MSE が「 P のみ」では 1.79×10^{-6} だったのに対し，「全変数」では 9.82×10^{-8} であった．「 P のみ」の条件では，一定期間後 $W(t)$ の値が大きく発散し正しく学習が進行しなかったのに対し，「全変数」では観測した 16000 ステップの間安定的に学習が進行した.^{*6}また「全変数」に関して 500 ステップ毎に $W(t)$ を採取し，それぞれの MLE および ESP Length の遷移を評価したところ，第6章で議論されたような安定状態と不安定状態の間の往来現象が観察された．これは次元削減された統合学習システムでも，もとの DIT の機能性が再現されていることを示唆する．

7.4 議論

本章では，これまでの章で提案された Input Reservoir, Neuralizing, DIT の手法を駆使・統合し，Innate Training そのものが表現されたニューラルネットワークである統合学習システムの設計手法が提案された．ここでは第1章で定義された要件と照らし合わせて，自律的な学習システムとしての統合学習システムの妥当性が議論される．まず構築された統合学習システムは，Chaotic ESN が入力との間のフィルタ，すなわち一種の Input Reservoir として自発的に活動しながら教師軌道を生成するため，高い自発性を有するといえる (要件 1)．すなわち仮にある入力がない状況でも，あるいは入力が付与された場合でも恒常的に学習が進行し停止し

^{*3} すなわち ESP Length が大きくなった．

^{*4} この閉ループは 50000 ステップの軌道をサンプリングし，Ridge 回帰によって達成された．

^{*5} 隠れノードのダイナミクスに関しては付録 E 中の図を参照せよ．

^{*6} この値の発散は，近似誤差の蓄積によって $W(t)$ が ELM の定義域から外れたことに起因すると考えられる．

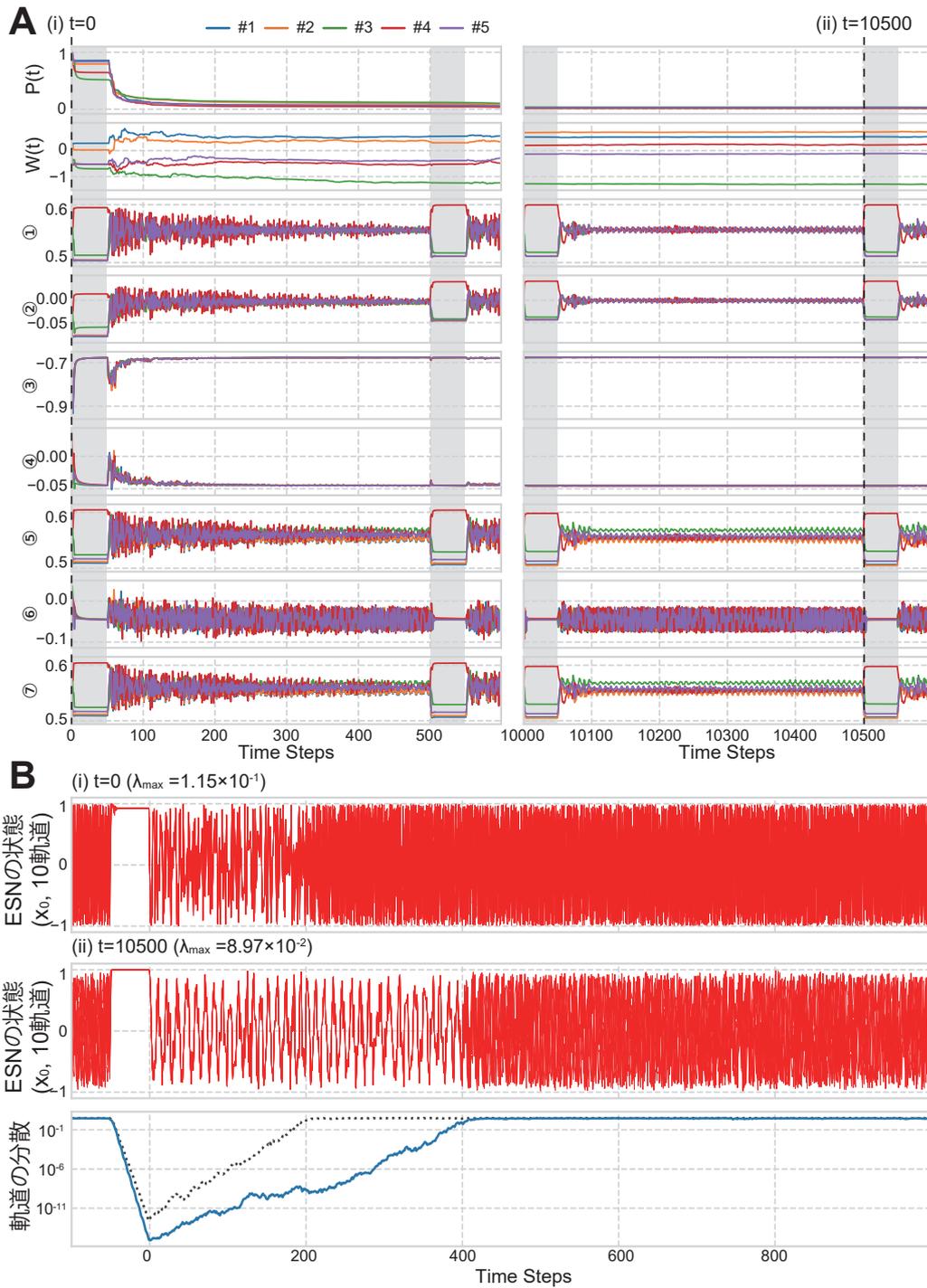


Fig. 7.2 統合学習システムのデモンストレーション。(A) 統合学習システムの各NISの軌道。灰色の領域で矩形波入力が入力される。凡例はNISの通し番号を表す各番号は図7.1内のELMの番号に対応し、その中の隠れ層の内の1ノードのダイナミクスが表示される。(B) 統合学習システムにおける学習の効果。得られた結合行列を用いてその入力特性を評価した。また対応するMLEを計測した。

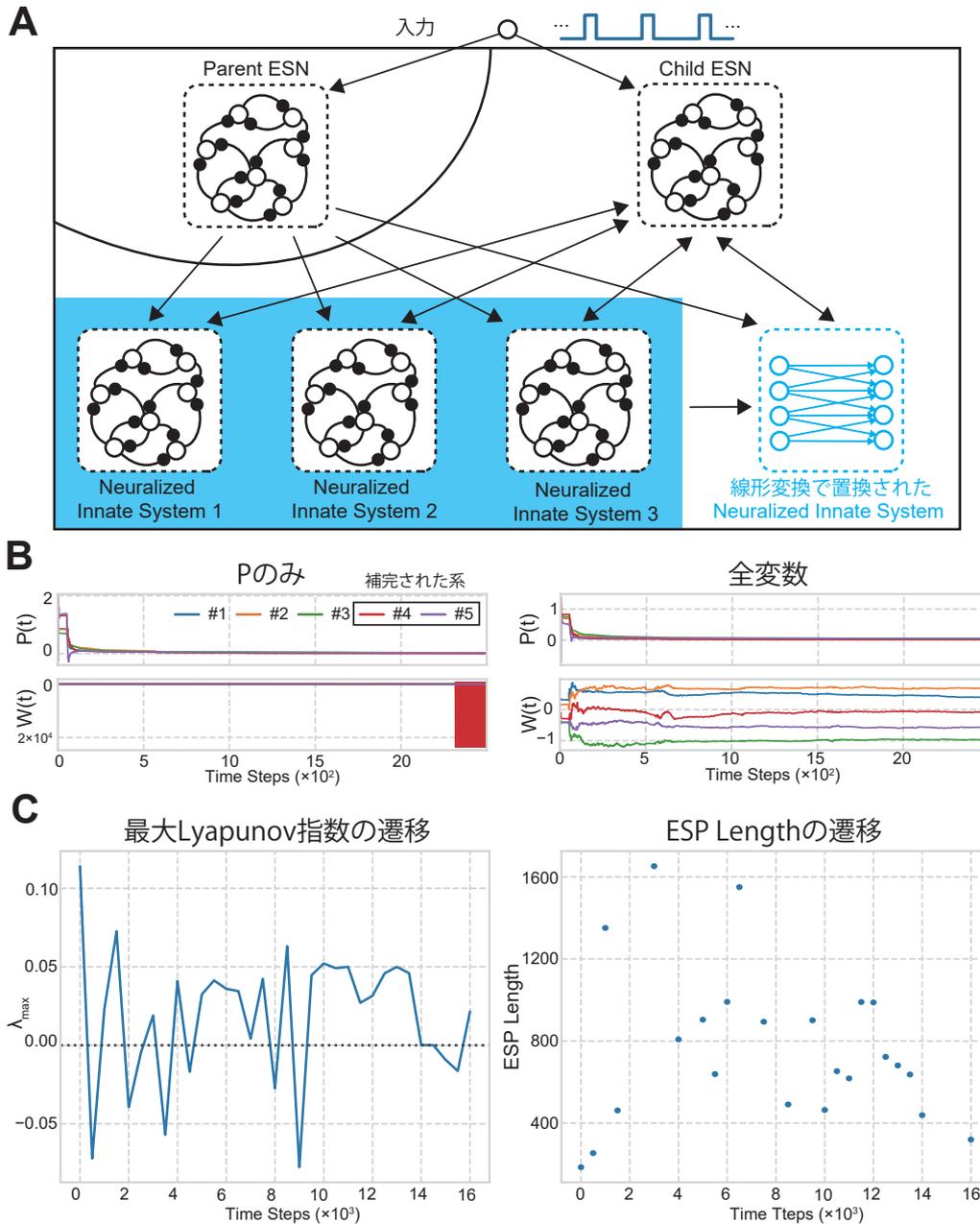


Fig. 7.3 AIOR (All-In-One Reservoir) を応用した統合学習システムの次元削減。(A) AIOR の手法を用いた NIS の置換。計 5 個ある NIS の 3 個の内部状態を用いて残り二つの $P(t)$ のダイナミクスを線形近似する。(B) 実際の学習中のダイナミクスの様子。閉ループを構成する上で、NIS 内の $P(t)$ のみを使う場合と、ELM の隠れ層を含めた全変数を使用する 2 種類が検証された。(C) 次元削減された系の妥当性の評価。第 6 章同様、MLE および ESP Length を用いて評価した。

ない。次に、通常の **Innate Training** とは異なり、教師軌道が **Parent ESN** のカオスダイナミクスとして生成され続けるため、この系は学習対象をより高い水準で構成しているといえる（要件 2）。また **Innate Training** で見られたようなハイパーパラメータ^{*7}やそれを保持する外的な記憶機構は不要で、全て内的なダイナミクスで実現されている。同様に **RLS** アルゴリズムの導入による **Optimizee**, **Optimizer** の分離も、**Neuralizing** の導入によって全て系内部の同一平面上の機構で克服されている。このように実現された統合学習システムは、その系の非線形性によって一様に学習に必要な機構を実現している（要件 3）。以上をまとめれば、本章で実現された統合学習システムは、自律的な学習機能がそのダイナミクスの上で一貫して実現された高次元非線形力学系、すなわち自律的な学習システムであるといえる。

また図 7.3 で示されたように、ニューラルネットワークとして表現された状態を計算資源と活用し、**AIOR** の発想を応用して、冗長な成分が取り除かれた洗練された統合学習システムの形態が得られることが期待される。現時点でも **tanh** で一貫しているという意味で自己充足的な構成であるが、依然として **NIS** や **Child ESN**, **Parent ESN** といったモジュールが中に確認される。このようなモジュールも、今後同様の操作で次元削減されることでその境界が曖昧となり、より学習システムの表現の一貫性の向上が期待される。

^{*7} 教師軌道の初期値等。

第 8 章

結論と展望

8.1 結論

本研究では、生物知能の重要な特徴でありかつ現在の人工知能で十分に達成されていない「自律的な学習機能」を実現する数理モデルの設計論を構築した。具体的には **Innate Training** と呼ばれる高次元カオス力学系を積極的に活用する機械学習手法を拡張し、自律的な学習機能を有する高次元非線形力学系の構成手法を提案した。

まず第 1 章において、自律的な学習システムが満たすべき 3 点の要件を、生物知能と既存の機械学習手法との比較を通して導入した。そして第 2 章では、この 3 点の要件を部分的に満足する数理モデルとして **Innate Training** が導入され、その特色と自律的な学習システムとしての適性が議論された。そして **Innate Training** がそのままでは自律的な学習システムの 3 要件を完全には満たさないことを指摘し、自律的な学習システム実現のための 4 点の改善点を策定した。第 3-6 章は各々の改善点に対応し、各章において新たな改善手法が提案された。第 3 章では **Input Reservoir** の概念を提唱し、**Innate Training** において重要な役割を果たす過渡的入力生成機構を、高次元非線形力学系内の部分系として表現できることを示した。この **Input Reservoir** により、外的な機構を **Input Reservoir** として系に内在化させる着想が可能となり、学習システムの表現に関してその一貫性を向上させる。第 4 章では、**Innate Training** で設計者によって制御されている入力の生成および切り替えを、**Input Reservoir** ならびに系が生成する高次元カオス軌道を活用し、線形閉ループを介して高い操作性で設計する手法を提案した。この提案手法を駆使することで、設計者からの一定の独立性が担保された、入力の生成および切り替えの自発的な構成が実現できることを示した。第 5 章では **AIOR** (**All-In-One Reservoir**) と呼ばれる概念と **Neuralizing** と名付けられる手法を提唱した。これらを統合し **Innate Training** の学習メカニズム、すなわち **RLS** アルゴリズムと **Readout** が内包された **RNN** である **NAIOR** (**Neuralized All-In-One Reservoir**) の構成方法を提案した。この **NAIOR** はその学習機能が、学習部と被学習部の乖離なく **RNN** 上に実現された高次元非線形力学系で、学習システムの自

己充足性の向上に大きく寄与する。第6章では、Innate Training において設計者によって制御される教師軌道の生成・保持機構そのものが系自身のダイナミクスによって実現される DIT (Developmental Innate Training) を提案した。そしてカオス力学系が Input Reservoir として片方向的に結合する構成において Innate Training と同様の学習の効果が確認されるのみならず、恒常的に学習の進行と崩壊が繰り返される自己組織化臨界現象状のダイナミクスが観測されることを示した。これらの結果は Input Reservoir としてカオス的な部分系が内包され片方向的に投射される構成が、恒常的な学習機能の実現において重要であることを意味し、自律的な学習システムが要請するネットワークポロジーに関する制約の存在を示唆する。第7章では、ここまでの Input Reservoir, Neuralizing, および DIT を統合し、自律的な学習機能が RNN 上のダイナミクスとして実現された「統合学習システム」を構築した。得られた統合学習システムは、まず Input Reservoir として Chaotic ESN を内包し教師軌道を生成し続けるので自発的である(要件1)。また DIT との統合で、Innate Training で見られたような教師軌道の設定(教師軌道の初期値等)やその保持機構は、全て RNN 内のダイナミクスで実現されている。その意味で、統合学習システムは学習対象に対する自己決定・構成能を有している(要件2)。さらに Neuralizing の活用によって RLS アルゴリズム等、Innate Training では外的な機構として存在した学習メカニズムが、全て RNN として系内部の同一平面上の機構で実現されている。したがって、得られた統合学習システムは自己充足的である(要件3)。また AIOR の着想を応用し、そのダイナミクスを計算資源として活用することで、学習機能を損なわずに次元を削減できることを示した。この次元削減は RNN 内の冗長な成分を活用したもので、RNN を構成しその情報処理能力が活用されることで初めて実現される操作といえる。そしてこの次元削減の操作により、統合学習システムのより洗練された構成が得られることが期待される。

8.2 今後の展望

まず第7章で示された AIOR をさらに適用することで冗長な成分の更なる削減が潜在的には可能である。例えば統合学習システムにおける、Parent ESN や Child ESN を構成するような閉ループも原理的には構築可能である。また第5章で指摘されたように、Neuralizing を用いて構築された現在の系は冗長な成分を多数含み、同様に次元削減の対象となりうる。このような次元削減の操作の果てに、自律的な学習機能を実現する高次元非線形力学系の最小構成単位が得られることが期待される。そしてこの最小構成単位のトポロジーないしは力学系的特性は、自律的な学習機能の実現における本質的かつ重要な制約条件を反映していると考えられる。^{*1}このように、本研究で得られた統合学習システムをさらに洗練化させ最小構成単位を獲

^{*1} 得られる最小構成単位は学習機能が封入された一種の“The Reservoir” [42] であるとみなせるだろう。そしてその動的特性は、学習機能を成立する上で重要な役割を果たしていることが期待される。自然言語の構造がうまく反映された力学系として ALBERT [43] の動的特性が解析されたが [42]、同様の着想がこの自律的な学習

得し、その動的特性を解析することが次の重要な研究課題となる。

本研究では、非線形力学系のダイナミクスの上で自律的な学習機能が一貫して実現する設計論が構成された。この設計論は、計算機上の数理モデルにとどまらず、物理系一般に適用可能である。この汎用性を活用し、実世界上でその環境との相互作用を通して学習する物質やロボット等の、実システムの実装に関する研究が今後展開されるものと期待される。そしてそのような実システムは一度構成されてしまえばそのパラメータの調整なしに、そのダイナミクスの上で学習が達成される。第5章でも議論されたように、このモデルパラメータの調整の不要性は、PRC (Physical Reservoir Computing) [51] やニューロモーフィックデバイスに代表される物理実装の文脈において効力を発揮するだろう。すなわち、古典計算機の制約に律速されず、その物理特性を最大限発揮する形で、自律的な学習機能を物理系に移植することが可能になる。こちらも本研究の延長線上にある重要な研究課題に位置付けられる。

また本研究では入力が開ループ系で与えられ学習し続ける統合学習システムを構築した。そのような学習システムは環境入力に対するカオス的な応答を常に獲得し続ける系であると解釈されるだろう。第1章でも議論された通り、生物も環境が変化しない状況でも動的に内部表象を更新し続ける自発的なシステムであり、本研究で得られた自律的な学習システムはその特性を上手く反映しているといえる。一方で生物は開ループ的だけでなく閉ループ的な状況、すなわち自ら能動的に環境と相互作用する中で学習を行うこともできる。本研究ではそのような閉ループ的な構成は検証されなかったが、身体を介して閉ループが構築されることで、そのような構成は即座に実現される。他方でどのような身体でも上手く機能するわけではなく、身体の構造すなわち学習システムと環境の間のインターフェースが適切に設計されて初めて意味のある学習が成立すると予想される。^{*2}このような学習に適した、あるいは学習機能の成立に要請される身体の制約に関する研究もまた今後の重要な課題である。系統的な解析が比較的容易な、Tensegrity系 [119, 120] 等の単純な身体がそのような研究の端緒として候補に挙がる。

また第4章で議論されたように、所定の記号を切り替えるだけでなく、究極的には自己組織的に記号を獲得するシステムの構成が課題となる。このような記号獲得システムを実現するには、得られた自律的な学習システムに加え、何かしらの教師なし学習の機構が必要となるが、その機構の開発も今後の重要な課題となる。また先述の議論同様、このような記号獲得システムも実環境中に置かれ、その相互作用によって実環境の情報が反映された記号が獲得されてはじめて有意義なものとなる。したがって、上記の実ロボットの実装と並行して、このような記号獲得システムの開発と研究が行われることが望ましいが、こちらも今後の課題に属する。

また物理系を介した閉ループ系に関する研究の帰結として、実世界上で環境に動的に相互作用しながら学習し続けるロボットの実現が期待される。本研究で提供されるのは、その内汎用

システムにおいても適用されうる。

^{*2} 生物は進化の過程でそのような「適切な」身体を獲得したと解釈できる。

的な「脳」の部分で、適切にインターフェースが設計されれば原理的にはどのロボットにも活用できる。また PRC や AIOR の思想を応用し、先ほど議論されたようにその学習機能を身体の要素で表現することも可能である。最終的には、身体・脳・環境の密な相互作用の上で学習し続ける自律的な学習システムの物理的実体を得られることが期待されるが、それはまさしく生物知能の形態そのものである。そしてそれこそが第 1 章の冒頭で議論された、人工知能分野が目指す究極的な AI の形態の一つの帰結とみなされるだろう。このように本研究の知見は、汎用人工知能の構成における礎を提供するもので、大きな価値を持つ。

付録 A

Reservoir Computing の数理的背景

UAP (Universal Approximation Property) について、ある機械学習モデルのクラスがどんな連続な関数も任意の精度で近似できる時、その機械学習モデルのクラスは UAP を有すると表現される。Horink ら [121] は 3 層の Feedforward Neural Network (FNN) に関する UAP を、活性化関数が Squashing Function^{*1} である場合において証明した。Leshno ら [122] はより一般的な活性化関数において 3 層の FNN の UAP を証明した。また一層の FNN と等価な ELM に関しても同様に UAP が証明されている。

ここまで紹介されたのは、時間的な構造のない入力機械学習モデルの UAP に関する研究である。一方で RNN のような時系列処理を行う機械学習モデルでは過去の入力時系列のフィルタ^{*2}の近似性能、すなわち UAP が問題となる。このような時間的な機械学習モデル^{*3}の UAP に関して、RC の文脈では特にフィルタ理論を基盤とした研究がなされてきた。その結果、Separation Property^{*4}を有するフィルタの集合^{*5}と UAP をもつ Readout を組み合わせた時、あらゆる Fading Memory Property^{*6}を有する時不変のフィルタが任意の精度で近似されることが証明されている [123, 124, 125]。極端な例としては、過去入力を完全に保持するシフトレジスタに FNN を Readout として取り付けた構成^{*7}が挙げられる。Maass らによって提案された Liquid State Machine [124] は UAP を有する時間的な機械学習モデルを形式化した概念であり、特に論文中では Spiking Neural Network に線形 Readout が取り付けられた構成の UAP が論文中で示されている。

フィルタ理論を基盤とした時間的な機械学習モデルの UAP の議論とは別に、ESP を導入し

*1 シグモイド関数状の有界非減少な関数。

*2 時系列 $u(\cdot)$ から時系列 $v(\cdot)$ への変換。

*3 すなわち力学系であり、RC の文脈では特に Reservoir と呼称される。

*4 点分離特性。任意の異なる二つの入力時系列を別々に表現するフィルタの特性を指す。

*5 フィルタバンク (Filter Bank) とも呼ばれる。

*6 減衰記憶特性。過去入力が時間発展により減衰する性質を指す。

*7 すなわち、記憶機構と非線形演算機構が完全に分離した系。

た UAP の議論がなされてきた。ESP (Echo State Property) はある入力での系の漸近状態収束の条件である。言い換えれば ESP を持つ系では、どのような初期状態から時間発展しても、ある入力時系列の下で漸近的に同じ状態に収束する。従って ESP を持つ系の内部状態は、定常的には過去の入力のある時不変な関数で記述される。ESN は ESP と Fading Memory Property を有する力学系の典型的な数理モデルであり、ESN と線形を組み合わせた構成の UAP が同様に証明されている [126, 127, 128]

付録 B

自発的な記号遷移 補足資料

B.1 Innate Training のアルゴリズムおよび詳細設定

Step 1 における結合行列 J はアルゴリズム 1 に書かれたアルゴリズムによって学習された。ここで A は調整される Chaotic ESN のノードの集合、 $B(i)$ はノード i に結合されるノードの添字集合を意味する。本研究では Chaotic ESN のノードの内半分が選択され学習された ($|A| = N_c/2$)。また $\Delta t = 1$ を採用した。

B.2 Chaotic ESN への投射行列の制御

バイアス項によって引き起こされる分岐によって Chaotic ESN のカオス性が失われることを防ぐため、記号遷移の後 0 に収束するような過渡的なダイナミクスが Chaotic ESN に投射されるように、Input ESN と Chaotic ESN をつなぐ結合行列 J^{ic} を調整した。

$$J^{ic} \mathbf{x}^{\text{in}}(t - t_{\text{switch}}) \approx \left[(t - t_{\text{switch}}) \cdot \exp\left(-\frac{1}{2} \left(\frac{t - t_{\text{switch}}}{50}\right)^2\right) \right] \mathbf{v}_s \quad (\text{B.1})$$

ここで $\mathbf{v}_s \in \mathbb{R}^{N^{\text{ch}}}$ (記号 s) は標準正規分布 $\mathcal{N}(0, 1)$ よりサンプルされた。

B.3 Innate Training 後の固有値分布の変化

図 B.1A は Innate Training による ESN の固有値の変化を示したものである ($M = 1$)。 L_{innate} が小さい時、スペクトル半径が大きく 1 を越えている様子が分かる。対称的に、 L_{innate} が大きい時固有値が単位円内に収まる様子が観察できる。

Algorithm 1 Innate Training のアルゴリズム

```

1: for  $i \in A$  do
2:    $P^i \leftarrow I$ 
3: for  $s \in S$  do
4:   initializing  $\mathbf{x}_{\text{target}}^s(0)$ 
5:   recording  $\mathbf{x}_{\text{target}}^s(t)$  ( $t \in [0, L_{\text{innate}})$ )
6: for epoch = 1 to 200 do
7:   for  $s \in S$  do
8:     initializing  $\mathbf{x}^s(0)$ 
9:      $t \leftarrow 0$ 
10:    count  $\leftarrow 0$ 
11:    while  $t < L_{\text{innate}}$  do
12:      for  $s \in S$  do
13:        if count = 0 (mod 2) then
14:          continue ▷ 冗長なサンプリングを避けるため導入される
15:           $\mathbf{e} \leftarrow \mathbf{x}^s(t) - \mathbf{x}_{\text{target}}^s(t)$ 
16:          for  $i \in A$  do
17:            for  $j \in B(i)$  do
18:               $J_{ij} \leftarrow J_{ij} - \mathbf{e}_i \sum_{k \in B(i)} P_{jk}^i \mathbf{x}_k^s(t)$ 
19:              for  $k \in B(i)$  do
20:                 $Q_{jk}^i \leftarrow P_{jk}^i - \frac{\sum_{m \in B(i)} \sum_{n \in B(i)} P_{jm}^i \mathbf{x}_m^s(t) \mathbf{x}_n^s(t) P_{nk}^i}{1 + \sum_{m \in B(i)} \sum_{n \in B(i)} \mathbf{x}_m^s(t) P_{mm}^i(t) \mathbf{x}_n^s(t)}$ 
21:                 $P^i \leftarrow Q^i$ 
22:             $t \leftarrow t + \Delta t$ 
23:            count  $\leftarrow$  count + 1

```

B.4 最大 Lyapunov 指数

MLE (最大 Lyapunov 指数) は島田・長島法 [129, 130] に基づいて計算された。なおアルゴリズム 2 において詳細な計算方法が記載されている。特に, $\Delta T = 1000$ ms, $T_{\text{horizon}} = 10^6$ ms, $l_{\text{pert}} = 10^{-6}$ の設定で 10 サンプル平均することで得た。

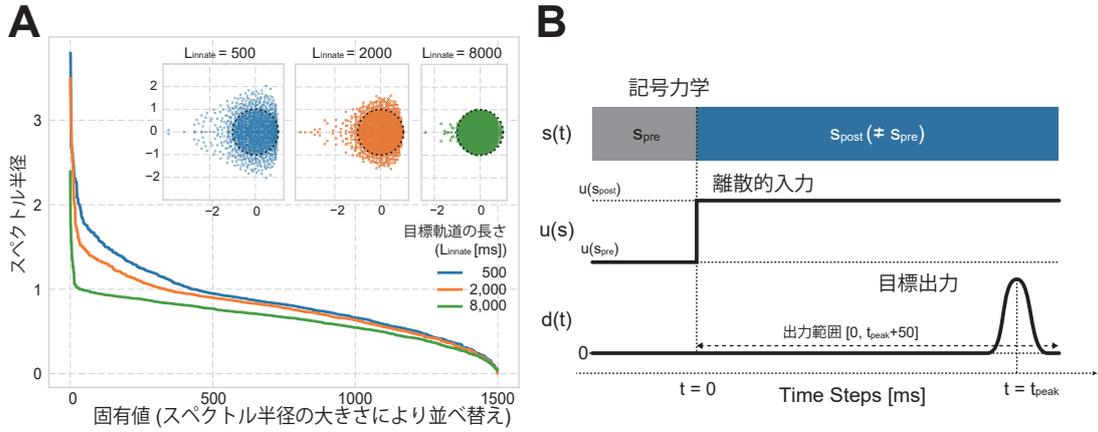


Fig. B.1 内部結合のスペクトル半径の解析および Timer Task の設定. (A) Innate Training 後の固有値の分布と J のスペクトル半径. innate trajectory の長さを変えて学習された 3 条件 ($L_{\text{innate}} = 500, 2000, 8000$) の結果が表示される. (B) Timer Task の概要. 第 3 章とは異なり離散的な入力が投射される. t_{peak} 後にピークが来るようなパルス波が入力される.

B.5 局所 Lyapunov 指数

入力記号 s に関する LLE (局所 Lyapunov 指数) は以下の式で定義される.

$$\text{LLE}(t) := \left\langle \log \left(\frac{\|\mathbf{x}_{\text{pert}}^s(t) - \mathbf{x}^s(t)\|}{\|\mathbf{x}_{\text{pert}}^s(0) - \mathbf{x}^s(0)\|} \right) \right\rangle \quad (\text{B.2})$$

ここで $\mathbf{x}^s(t)$ は入力記号 s に対する内部ダイナミクス ($t = 0$ で入力が切り替わる設定), $\mathbf{x}_{\text{pert}}^s(t)$ は, 元の軌道 $\mathbf{x}^s(t)$ に時刻 $t = 0$ ms で摂動を与えたものである.*1 ブラケットはサンプルに対する平均を表す. 特に 10 サンプルを用いて計算された.

B.6 長周期の周期的遷移の設計

提案手法と従来手法の操作性を比較するため, 10 個の長周期の周期的遷移を用意しその埋め込み精度を, FORCE 学習と提案手法に関して評価した. 具体的には, k セグメントの周期的曲線 $[x_k(t), y_k(t)]^T$ を T_{switch} 毎に切り替えることで以下の目標軌道を得る:

$$x_k(t) := \cos \left(\frac{2\pi}{k} \left\lfloor \frac{t}{T_{\text{switch}}} \right\rfloor \right) + 0.25 \sin \left(\frac{8\pi t}{T_{\text{switch}}} \right) \quad (\text{B.3})$$

$$y_k(t) := \sin \left(\frac{2\pi}{k} \left\lfloor \frac{t}{T_{\text{switch}}} \right\rfloor \right) + 0.25 \sin \left(\frac{12\pi t}{T_{\text{switch}}} \right) \quad (\text{B.4})$$

*1 $\mathbf{x}_{\text{pert}}^s(0) = \mathbf{x}^s(0) + \epsilon, \|\epsilon\| = 10^{-6}$

Algorithm 2 最大 Lyapunov 指数の計測

-
- 1: initializing $\mathbf{x}(0) \in \mathbb{R}^N$
 - 2: sampling $\boldsymbol{\epsilon} \in \mathbb{R}^N$ ($\boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$)
 - 3: $\boldsymbol{\epsilon} \leftarrow l_{\text{pert}} \frac{\boldsymbol{\epsilon}}{\|\boldsymbol{\epsilon}\|}$
 - 4: initializing $\mathbf{y}(0) := \mathbf{x}(0) + \boldsymbol{\epsilon}$
 - 5: $t \leftarrow 0$
 - 6: Lyap = $\{\phi\}$
 - 7: **while** $t < T_{\text{horizon}}$ **do**
 - 8: Lyap \leftarrow Lyap $\cup \left\{ \frac{1}{\Delta T} \log \left(\frac{\|\mathbf{y}(t+\Delta T) - \mathbf{x}(t+\Delta T)\|}{\|\mathbf{y}(t) - \mathbf{x}(t)\|} \right) \right\}$
 - 9: $t \leftarrow t + \Delta t$
 - 10: $\mathbf{y}(t) \leftarrow \mathbf{x}(t) + l_{\text{pert}} \frac{\mathbf{y}(t) - \mathbf{x}(t)}{\|\mathbf{y}(t) - \mathbf{x}(t)\|}$
 - 11: calculating MLE by averaging values in Lyap
-

本研究では T_{switch} を 1000 ms から 1500 ms に、 k を 1 から 5 に変化させることで累計 10 個の目標軌道を生成した。

FORCE 学習の評価に関して、1500 次元の chaotic ESN と二次元の線形閉ループを以下の式のように用意した。

$$\tau \dot{\mathbf{x}} = -\mathbf{x} + \tanh(gJ\mathbf{x}(t) + \mathbf{U}_{\text{feed}}\mathbf{z}(t)) \quad (\text{B.5})$$

$$\mathbf{z}(t) = w\mathbf{x}(t) \quad (\text{B.6})$$

なお $\tau = 10$, $g = 1.5$ の値を採用し、 $J \in \mathbb{R}^{1500 \times 1500}$ を標準正規分布 $\mathcal{N}(0, 1/1500)$ より生成した。 $\mathbf{U}_{\text{feed}} \in \mathbb{R}^{1500 \times 2}$ はランダムなフィードバック結合で一様乱数 $\mathbf{U}([-1, 1])$ から生成された。FORCE 学習を用いて $w \in \mathbb{R}^{2 \times 1500}$ が学習される。 $10kT_{\text{switch}}$ [ms] にわたって FORCE 学習によって w を調整し、直後 $5kT_{\text{switch}}$ [ms] の出力を目標軌道との NMSE によって評価した。

提案手法を評価するため、Innate Training によって事前学習済みの ESN ($(M, L_{\text{innate}}) = (5, 2000)$) を使用した。また目標軌道 $[x_k(t), y_k(t)]^T$ を元の k 個のセグメントに分割し、 f_{max} を T_{switch} 毎に記号を切り替わるように、そして Readout w_{out} を対応するリサージュ曲線を書くように学習させた。1 セグメントの時は記号の切り替えが生じないため、2 記号間の遷移を f_{max} の学習で埋め込み、 w_{out} を同じリサージュ曲線を出力させるように学習させた。そして FORCE 学習同様、 $5kT_{\text{switch}}$ [ms] にわたる軌道より精度を評価した。

図 B.2 は FORCE 学習と提案手法のそれぞれに対する埋め込み精度であり、特にセグメントが複数ある ($k \geq 2$) 状況で提案手法が FORCE 学習の誤差を大きく下回った。記号を先行的に与え分割することにより、提案手法を用いて長い周期のダイナミクスの埋め込みが可能となる。

B.7 履歴依存的な確率遷移則の埋め込み

さらにカオス軌道の高次元性を活用することで、履歴依存的な確率遷移則を埋め込むことを示す。実験では図 B.3A で示されるような単純なマルコフモデルでは表現できない有限次元の履歴依存的な遷移則を用意した。ESN としては $(M, L_{\text{innate}}) = (3, 1000)$ で事前学習されたものを使用した。図 B.3B は記号力学, ESN のダイナミクスおよび出力軌道を、10 個の微小摂動を与えて時間発展させて重ね描かれている。摂動を与えてから 30000 [ms] 後、遷移がばらつく様子が観察されたがいずれのサンプルに関しても、定義された遷移則に沿った記号遷移が生み出されている。図 B.3C は得られた軌道から遷移確率および滞在期間を評価したものであり、目標遷移則との差はあるものの記号 A における確率的な切り替えを高い精度で表現できている。このように本提案手法は、マルコフモデルで単純に評価できないような履歴依存的な確率遷移則をも設計することができる。

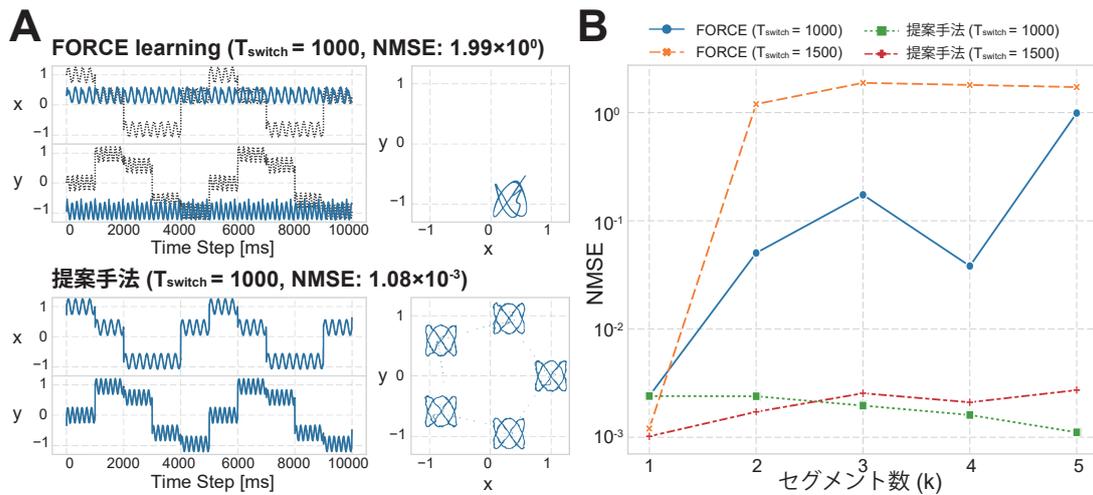


Fig. B.2 FORCE 学習と提案手法との設計性の比較。(A) 周期的な関数の学習例。点線は目標出力。(B) 埋め込みの精度切り替えの期間と記号の数を変えた系 10 個 (2×5) の目標遷移則を用意した。

B.8 軌道の履歴依存性の評価

カオス的遍歴の研究において、擬似アトラクタ間の遷移の仕方において、系特有の「嗜好」が存在する [113, 112] ことが指摘されている。すなわちその遷移の仕方は履歴依存的であり、その意味でランダムな遷移とは区別される。本研究で埋め込まれた確率的遷移も決定論的力学系とそのカオス性の上で成立している以上、同様の傾向が確認できる可能性がある。そこで本

研究では埋め込まれた確率遷移則とその軌道に関して、前の記号に応じた出力の変化を評価した。図 B.4A は遷移後 2500 [ms] から 2505 [ms] 間の出力軌道を表し、現在の記号とひとつ前の記号の違いに応じて色分けされている。同じ記号の状態にいるにもかかわらず、前の記号に応じて別の領域に分布する様子が観察されるが、これは得られた遷移の仕方に関しても履歴に依存するような「嗜好」を有することを示唆する。

同様に本提案手法がマクロ的な記号遷移に関して「ランダム性」を上手く表現できる、すなわち記号遷移の履歴だけでは将来の挙動を予測することが実質的に困難であることを示す。図 B.4B では確率遷移則 1, 2 に関して前の記号入力を含めた遷移行列を表示されている。多少の確率の偏りはみられるが、依然として値が前の記号に関係なくばらついており、その挙動の予測が単純に前の記号を観測するだけでは困難であることを意味する。まとめると ESN の軌道に関しては前の状態に応じた偏りが観察されたが、大域的な記号遷移の挙動は確率性を上手く提案手法は模倣できるといえる。

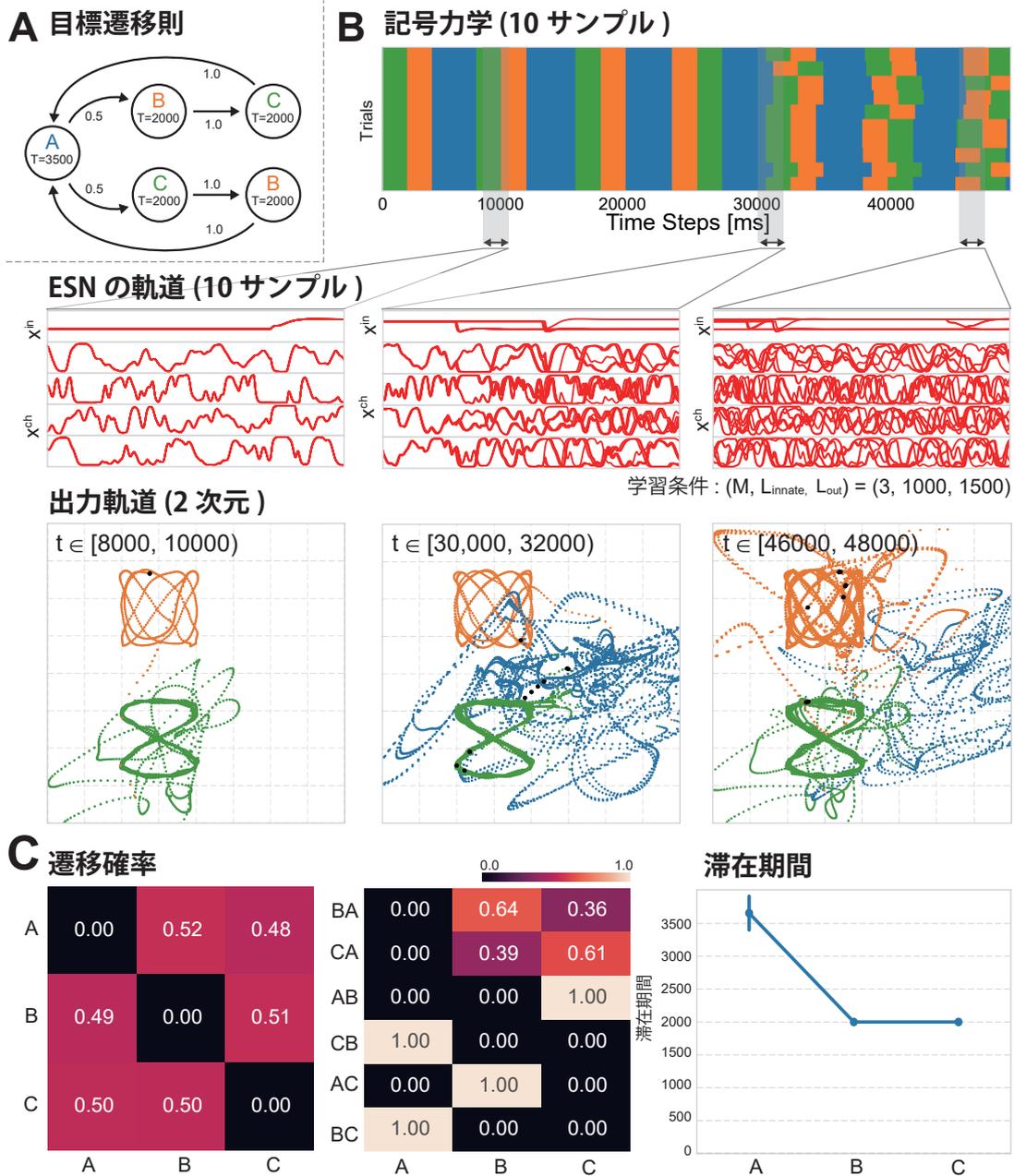


Fig. B.3 履歴依存的な確率遷移の埋め込み (A) 目標となる確率遷移則. B および C において前の記号に応じて出力を変えなければならない. (B) ESN のダイナミクスおよび対応する出力軌道. $(M, L_{\text{innate}}) = (3, 1000)$ の条件で学習された ESN を使用し, 目標軌道に基づき学習された f_{max} を使用し閉ループ系を構築した. 10 個の軌道が重ね描かれている. (C) 遷移の埋め込み精度の評価. 確率遷移行列と, 記号の切り替え期間が表示される.

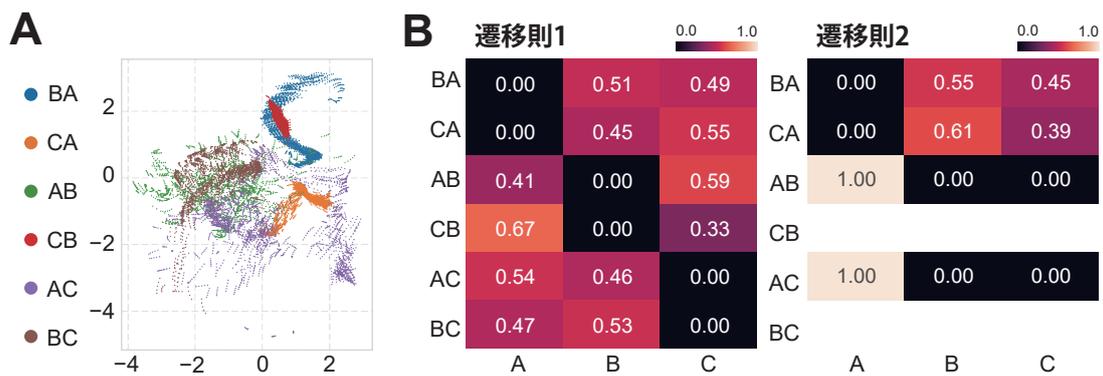


Fig. B.4 軌道の履歴依存性の解析. (A) 2500 から 2505 ステップにおける出力ダイナミクスの表示. 現在と直前の記号に応じて色付けがされている. (B) 過去 2 記号に関する確率遷移行列.

付録 C

Neutralized All-In-One Reservoir 補足 資料

[特許出願準備中のため非公表]

付録 D

Developmental Innate Training 補足 資料

D.1 他条件における解析結果

他の (μ, K, p) のパラメータに関しても本論と同様にスケールフリー性の解析図 D.1, ならびに結合行列の解析図 D.2 が行われた.

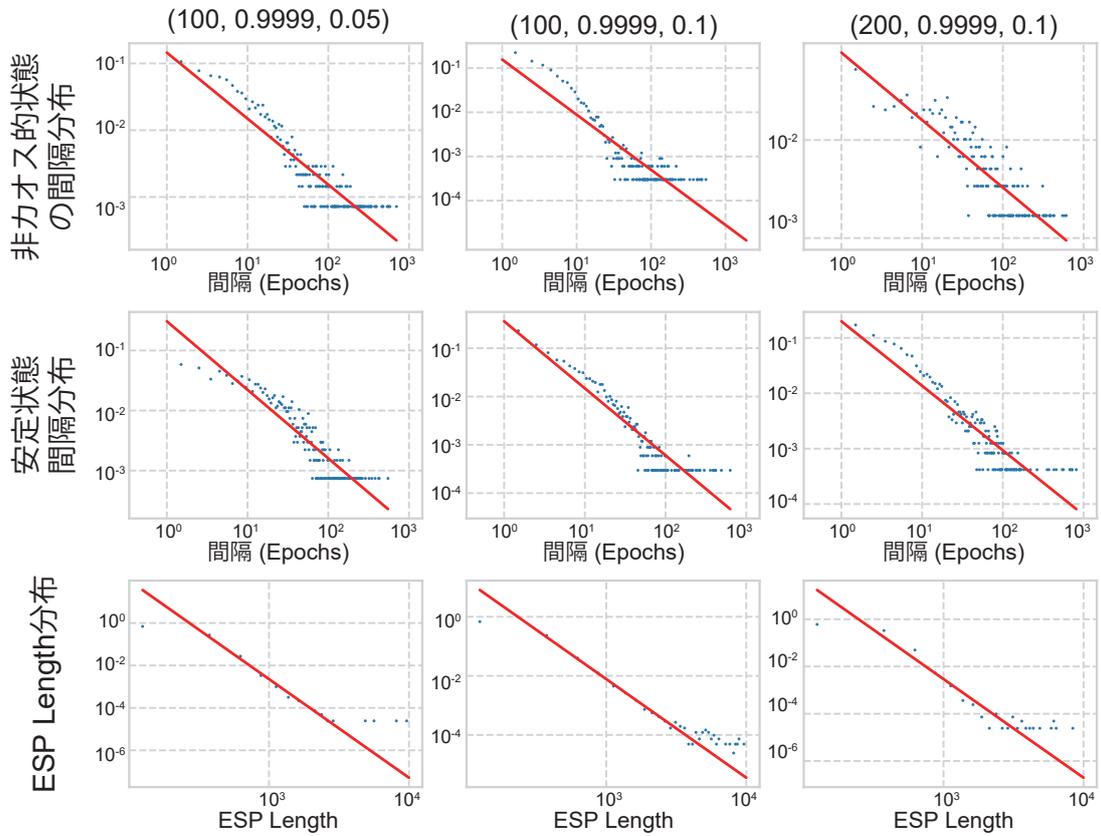


Fig. D.1 他条件におけるスケールフリー性の解析.

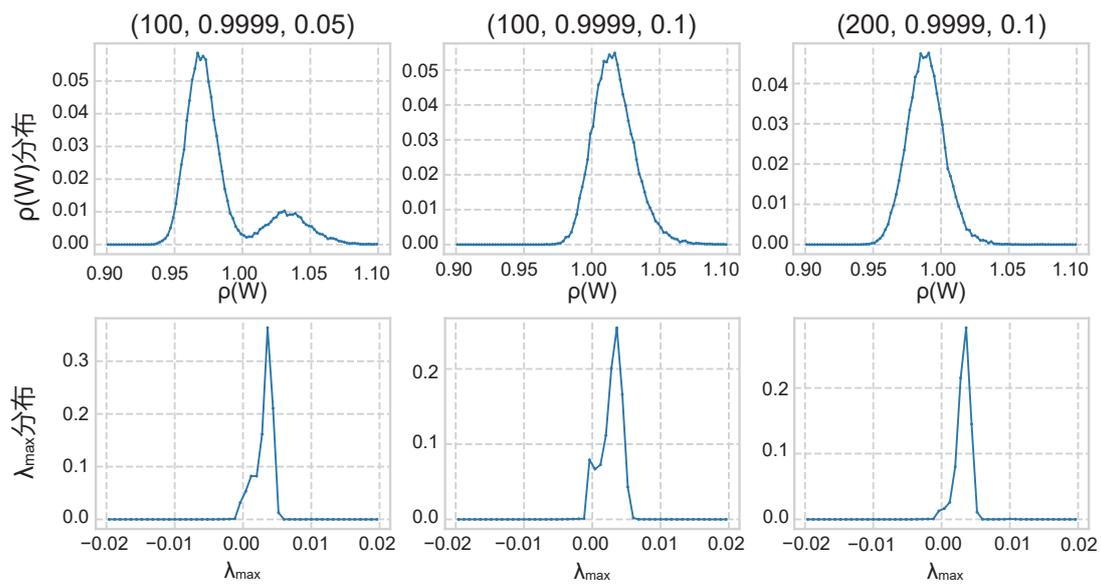


Fig. D.2 他条件における結合行列の解析.

付録 E

統合学習システム 補足資料

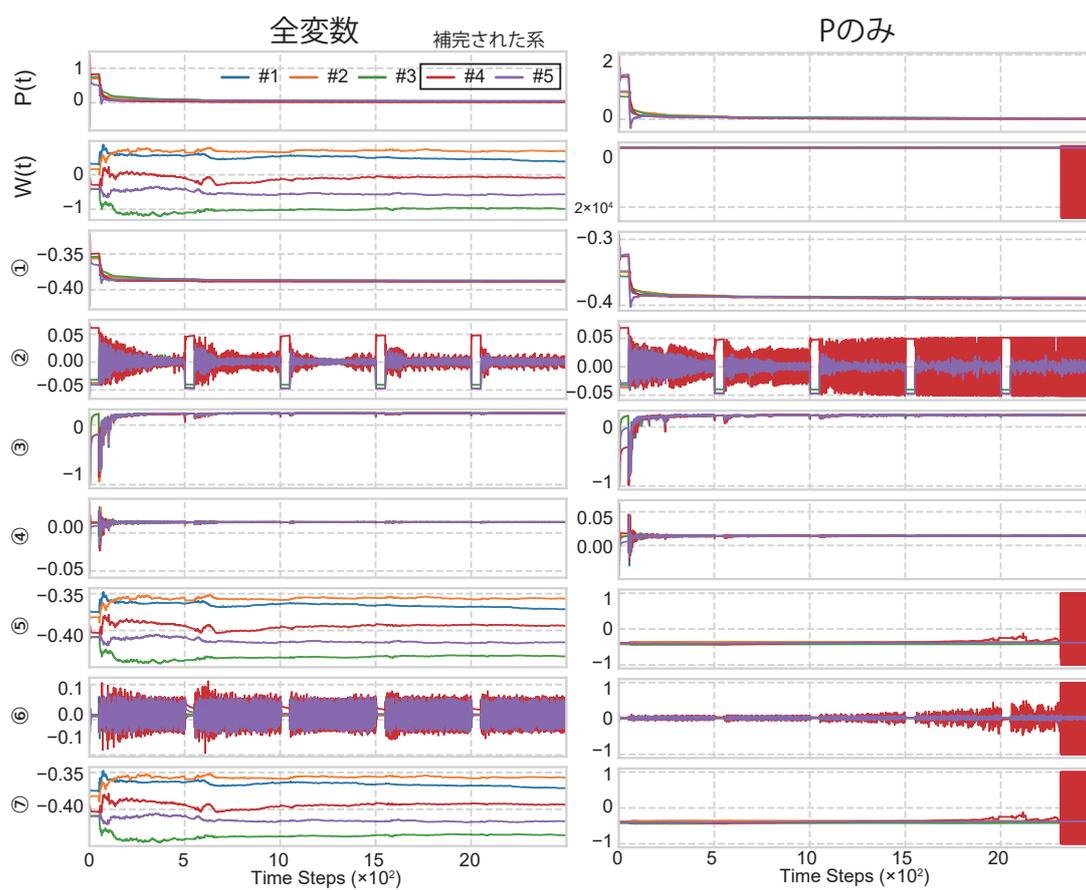


Fig. E.1 次元圧縮された統合学習システムの全ダイナミクス。図中の番号は、図 7.1B の NIS 内の ELM の番号に対応する。

謝辞

本論文は、筆者が東京大学 大学院情報理工学系研究科 知能機械情報学専攻在学中に、所属する知能システム情報学研究室での研究成果をまとめたものです。また本論文の一部の研究は JSPS 科研費 JP20J12815 の助成を受けて行われました。

同研究室 國吉康夫教授には、指導教員として本研究の実施の機会を与えていただきました。そして学部生の頃、筆者が研究室に配属された時より、筆者が携わり発表した研究全てに関して、終始一貫して厳しくも温かいご指導とご鞭撻を頂きました。ここに深謝の意を表します。

同研究科附属 情報理工学教育研究センター 中嶋浩平准教授には、研究構想、データ解析、論文執筆等の技術的な事のみならず、研究者としての心構えや研究そのものの真髄に至るまで非常に多岐に至る面で、熱心なご指導を賜りました。本当に感謝の念に堪えません。

同研究室 新山龍馬講師には、日頃の研究会での議論を通して、研究の意義や方向性に関して多く助言を賜りました。特にソフトロボティクスの専門家としてのコメントは、私の研究を補強する上で大変参考になるものが多く、非常に勉強になりました。厚く感謝の意を表します。

中部大学 創発学術院大学院 津田一郎教授、ならびに同専攻 高橋宏知准教授には副査として本論文を審査していただきました。そしてその過程で本論文の構成に関して、数多くの重要かつ有意義なご意見とご指導をいただきました。ここに深謝いたします。

同研究室 西川鋭助教 (現九州大学 工学研究院機械工学部門 准教授) には、日々の研究会や発表において数多くの有意義な助言をいただきました。また研究のみならず、研究室での生活に関して親身になって相談に乗っていただきました。この場で厚くお礼申し上げます。

同研究室 金沢星慶特任助教には日頃よりサブミーティングを通して、研究や発表内容に関して多くの温かい助言をいただきました。また事務的な作業や手続きに関して、数多く手助けをしていただきました。ここに感謝の意を表します。

また、その他ここに記しきれない数多くの方々のご支援によって、本研究は遂行されました。そのことを銘記しつつ、これまで携わった全ての皆様にこの場で深く感謝の意を申し上げます。本当にありがとうございました。

最後にこの博士課程における生活を通して惜しみない支援をしていただいた父と母に、心からの感謝を伝えたいと思います。

参考文献

- [1] Rolf Pfeifer and Christian Scheier. *Understanding intelligence*. MIT press, 2001.
- [2] Ludwig Von Bertalanffy. General systems theory. *The Science of Synthesis: Exploring the Social Implications of General Systems Theory*, Vol. 103, , 2010.
- [3] Norbert Wiener. *Cybernetics or Control and Communication in the Animal and the Machine*. MIT press, 2019.
- [4] Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, Vol. 27, No. 3, pp. 379–423, 1948.
- [5] 金子邦彦, 津田一郎. 複雑系のカオスのシナリオ, 第 1 巻. 朝倉書店, 1996.
- [6] 金子邦彦, 池上高志, 津田一郎編集. 複雑系の進化的シナリオ—生命の発展様式, 第 2 巻. 朝倉書店, 1996.
- [7] 井庭崇, 福原義久. 複雑系入門: 知のフロンティアへの冒険. NTT 出版, 1998.
- [8] 金子邦彦. 生命とは何か: 複雑系生命科学へ. 東京大学出版会, 2009.
- [9] 金子邦彦. 普遍生物学: 物理に宿る生命, 生命の紡ぐ物理. 東京大学出版会, 2019.
- [10] W Thomas Miller, Paul J Werbos, and Richard S Sutton. *Neural networks for control*. MIT press, 1995.
- [11] Vincent François-Lavet, Peter Henderson, Riashat Islam, Marc G Bellemare, and Joelle Pineau. An introduction to deep reinforcement learning. *arXiv preprint arXiv:1811.12560*, 2018.
- [12] Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. A brief survey of deep reinforcement learning. *arXiv preprint arXiv:1708.05866*, 2017.
- [13] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- [14] 佐々木智丈, 加嶋健司. 制御工学者のための強化学習入門. 計測と制御, Vol. 58, No. 3, pp. 182–188, 2019.
- [15] János Neumann, Arthur W Burks, et al. *Theory of self-reproducing automata*, Vol. 1102024. University of Illinois press Urbana, 1966.

- [16] Martin Gardner. Mathematical games. *Scientific american*, Vol. 222, No. 6, pp. 132–140, 1970.
- [17] Peter Gács. Reliable cellular automata with self-organization. *Journal of Statistical Physics*, Vol. 103, No. 1, pp. 45–267, 2001.
- [18] Humberto R Maturana and Francisco J Varela. *Autopoiesis and cognition: The realization of the living*, Vol. 42. Springer Science & Business Media, 2012.
- [19] H.R. マトゥラーナ, F.J. ヴァレラ, 河本英夫訳. オートポイエーシス — 生命システムとは何か. 国文社, 1991.
- [20] 郡司ペギオ-幸夫. 原生計算と存在論的観測—生命と時間、そして原生. 東京大学出版会, 2004.
- [21] 郡司ペギオ-幸夫, O.E. レスラー, 松野孝一郎. 内部観測. 青土社, 1997.
- [22] Christopher M Bishop. Pattern recognition. *Machine learning*, Vol. 128, No. 9, 2006.
- [23] Valentino Braitenberg. *Vehicles: Experiments in synthetic psychology*. MIT press, 1986.
- [24] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.
- [25] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, Vol. 550, No. 7676, pp. 354–359, 2017.
- [26] Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, Vol. 356, No. 6337, pp. 508–513, 2017.
- [27] John R Searle. Minds, brains, and programs. *Behavioral and brain sciences*, Vol. 3, No. 3, pp. 417–424, 1980.
- [28] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, Vol. 114, No. 13, pp. 3521–3526, 2017.
- [29] Walter J Freeman. Simulation of chaotic eeg patterns with a dynamic model of the olfactory system. *Biological cybernetics*, Vol. 56, No. 2-3, pp. 139–150, 1987.
- [30] Philippe Faure and Henri Korn. Is there chaos in the brain? i. concepts of nonlinear dynamics and methods of investigation. *Comptes Rendus de l'Académie des Sciences-Series III-Sciences de la Vie*, Vol. 324, No. 9, pp. 773–793, 2001.

-
- [31] Henri Korn and Philippe Faure. Is there chaos in the brain? ii. experimental evidence and related models. *Comptes rendus biologiques*, Vol. 326, No. 9, pp. 787–840, 2003.
- [32] Walter J Freeman. Evidence from human scalp electroencephalograms of global chaotic itinerancy. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, Vol. 13, No. 3, pp. 1067–1077, 2003.
- [33] Erol Başar. *Chaos in Brain Function: Containing Original Chapters by E. Basar and TH Bullock and Topical Articles Reprinted from the Springer Series in Brain Dynamics*. Springer Science & Business Media, 2012.
- [34] W.J. フリーマン, 浅野孝雄訳, 津田一郎校閲. 脳はいかにして心を創るのか: 神経回路のカオスが生み出す志向性・意味・自由意志. 産業図書, 2011.
- [35] 津田一郎. 脳の中に数学を見る. 共立出版, 2016.
- [36] Hatsuo Hayashi, Satoru Ishizuka, Masahiro Ohta, and Kazuyoshi Hirakawa. Chaotic behavior in the onchidium giant neuron under sinusoidal stimulation. *Physics Letters A*, Vol. 88, No. 8, pp. 435–438, 1982.
- [37] Kazuyuki Aihara, Gen Matsumoto, and Yuhji Ikegaya. Periodic and non-periodic responses of a periodically forced hodgkin-huxley oscillator. *Journal of theoretical biology*, Vol. 109, No. 2, pp. 249–269, 1984.
- [38] Dean V Buonomano and Wolfgang Maass. State-dependent computations: spatiotemporal processing in cortical networks. *Nature Reviews Neuroscience*, Vol. 10, No. 2, pp. 113–125, 2009.
- [39] Rafael Yuste. From the neuron doctrine to neural networks. *Nature reviews neuroscience*, Vol. 16, No. 8, pp. 487–497, 2015.
- [40] Emma R Wood, Paul A Dudchenko, R Jonathan Robitsek, and Howard Eichenbaum. Hippocampal neurons encode information about different types of memory episodes occurring in the same location. *Neuron*, Vol. 27, No. 3, pp. 623–633, 2000.
- [41] Edward Ott, Celso Grebogi, and James A Yorke. Controlling chaos. *Physical review letters*, Vol. 64, No. 11, p. 1196, 1990.
- [42] Katsuma Inoue, Soh Ohara, Yasuo Kuniyoshi, and Kohei Nakajima. Transient chaos in bert. *arXiv preprint arXiv:2106.03181*, 2021.
- [43] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.
- [44] Tamás Tél. Transient chaos. *Directions in chaos*, Vol. 3, pp. 149–211, 1990.
- [45] Howard Eichenbaum. The hippocampus and declarative memory: cognitive mechanisms and neural codes. *Behavioural brain research*, Vol. 127, No. 1-2, pp. 199–207, 2001.

- [46] Frank Haist, Jane Bowden Gore, and Hui Mao. Consolidation of human memory over decades revealed by functional magnetic resonance imaging. *Nature neuroscience*, Vol. 4, No. 11, pp. 1139–1145, 2001.
- [47] Rodrigo Laje and Dean V Buonomano. Robust timing and motor patterns by taming chaos in recurrent neural networks. *Nature neuroscience*, Vol. 16, No. 7, p. 925, 2013.
- [48] Artur Luczak, Peter Barthó, and Kenneth D Harris. Spontaneous events outline the realm of possible sensory responses in neocortical populations. *Neuron*, Vol. 62, No. 3, pp. 413–425, 2009.
- [49] Adrian Thompson. An evolved circuit, intrinsic in silicon, entwined with physics. In *International Conference on Evolvable Systems*, pp. 390–405. Springer, 1996.
- [50] Herbert Jaeger. The echo state approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, Vol. 148, No. 34, p. 13, 2001.
- [51] Kohei Nakajima. Physical reservoir computing an introductory perspective. *Japanese Journal of Applied Physics*, Vol. 59, No. 6, p. 060501, 2020.
- [52] Paul J Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, Vol. 78, No. 10, pp. 1550–1560, 1990.
- [53] Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, Vol. 1, No. 2, pp. 270–280, 1989.
- [54] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, Vol. 304, No. 5667, pp. 78–80, 2004.
- [55] David Sussillo and Larry F Abbott. Generating coherent patterns of activity from chaotic neural networks. *Neuron*, Vol. 63, No. 4, pp. 544–557, 2009.
- [56] Jean Ginibre. Statistical ensembles of complex, quaternion, and real matrices. *Journal of Mathematical Physics*, Vol. 6, No. 3, pp. 440–449, 1965.
- [57] Izzet B Yildiz, Herbert Jaeger, and Stefan J Kiebel. Re-visiting the echo state property. *Neural networks*, Vol. 35, pp. 1–9, 2012.
- [58] Gandhi Manjunath and Herbert Jaeger. Echo state property linked to an input: Exploring a fundamental characteristic of recurrent neural networks. *Neural computation*, Vol. 25, No. 3, pp. 671–696, 2013.
- [59] Marc Massar and Serge Massar. Mean-field theory of echo state networks. *Physical Review E*, Vol. 87, No. 4, p. 042809, 2013.
- [60] David Verstraeten, Benjamin Schrauwen, Michiel Haene, Dirk Stroobandt. An experimental unification of reservoir computing methods. *Neural networks*, Vol. 20, No. 3, pp. 391–403, 2007.

-
- [61] Jack Sherman and Winifred J Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *The Annals of Mathematical Statistics*, Vol. 21, No. 1, pp. 124–127, 1950.
- [62] Ruihong Wu, Kohei Nakajima, and Yongping Pan. Performance improvement of force learning for chaotic echo state networks. In *International Conference on Neural Information Processing*, pp. 262–272. Springer, 2021.
- [63] Kenji Matsumoto and Ichiro Tsuda. Noise-induced order. *Journal of Statistical Physics*, Vol. 31, No. 1, pp. 87–106, 1983.
- [64] Brian DePasquale, Christopher J Cueva, Kanaka Rajan, LF Abbott, et al. full-force: A target-based method for training recurrent networks. *PloS one*, Vol. 13, No. 2, p. e0191527, 2018.
- [65] Stefano Fusi, Earl K Miller, and Mattia Rigotti. Why neurons mix: high dimensionality for higher cognition. *Current opinion in neurobiology*, Vol. 37, pp. 66–74, 2016.
- [66] Katsuma Inoue, Kohei Nakajima, and Yasuo Kuniyoshi. Soft bodies as input reservoir: role of softness from the viewpoint of reservoir computing. In *2019 International Symposium on Micro-NanoMechatronics and Human Science (MHS)*, pp. 1–7. IEEE, 2019.
- [67] Vishwa Goudar and Dean V Buonomano. Encoding sensory and motor patterns as time-invariant trajectories in recurrent neural networks. *Elife*, Vol. 7, p. e31134, 2018.
- [68] Jannis Schuecker, Sven Goedeke, and Moritz Helias. Optimal sequence memory in driven random networks. *Physical Review X*, Vol. 8, No. 4, p. 041029, 2018.
- [69] Kohei Nakajima, Tao Li, and Nozomi Akashi. Soft timer: dynamic clock embedded in soft body. In *Robotic Systems and Autonomous Platforms*, pp. 181–196. Elsevier, 2019.
- [70] Daniela Rus and Michael T Tolley. Design, fabrication and control of soft robots. *Nature*, Vol. 521, No. 7553, p. 467, 2015.
- [71] Cecilia Laschi, Barbara Mazzolai, and Matteo Cianchetti. Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Sci. Robot.*, Vol. 1, No. 1, p. 3690, 2016.
- [72] Qian Zhao, Kohei Nakajima, Hidenobu Sumioka, Helmut Hauser, and Rolf Pfeifer. Spine dynamics as a computational resource in spine-driven quadruped locomotion. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 1445–1451. IEEE, 2013.
- [73] Kohei Nakajima, Helmut Hauser, Rongjie Kang, Emanuele Guglielmino, Darwin G Caldwell, and Rolf Pfeifer. Computing with a muscular-hydrostat system. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 1504–1511. IEEE, 2013.
- [74] Kohei Nakajima, Helmut Hauser, Rongjie Kang, Emanuele Guglielmino, Darwin G Caldwell, and Rolf Pfeifer. A soft body as a reservoir: case studies in a dynamic model of

- octopus-inspired soft robotic arm. *Frontiers in Computational Neuroscience*, Vol. 7, p. 91, 2013.
- [75] Kohei Nakajima, Tao Li, Helmut Hauser, and Rolf Pfeifer. Exploiting short-term memory in soft body dynamics as a computational resource. *Journal of The Royal Society Interface*, Vol. 11, No. 100, 2014.
- [76] Kohei Nakajima, Helmut Hauser, Tao Li, and Rolf Pfeifer. Information processing via physical soft body. *Scientific Reports*, Vol. 5, p. 10487, 2015.
- [77] Kohei Nakajima, Helmut Hauser, Tao Li, and Rolf Pfeifer. Exploiting the dynamics of soft materials for machine learning. *Soft robotics*, Vol. 5, No. 3, pp. 339–347, 2018.
- [78] Katsuma Inoue, Kohei Nakajima, and Yasuo Kuniyoshi. Designing spontaneous behavioral switching via chaotic itinerancy. *Science advances*, Vol. 6, No. 46, p. eabb3989, 2020.
- [79] Jun Namikawa and Jun Tani. A model for learning to segment temporal sequences, utilizing a mixture of rnn experts together with adaptive variance. *Neural Networks*, Vol. 21, No. 10, pp. 1466–1475, 2008.
- [80] Jun Namikawa and Jun Tani. Learning to imitate stochastic time series in a compositional way by chaos. *Neural Networks*, Vol. 23, No. 5, pp. 625–638, 2010.
- [81] Jun Namikawa, Ryunosuke Nishimoto, and Jun Tani. A neurodynamic account of spontaneous behaviour. *PLoS computational biology*, Vol. 7, No. 10, p. e1002221, 2011.
- [82] Yoshua Bengio, Patrice Simard, Paolo Frasconi, et al. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, Vol. 5, No. 2, pp. 157–166, 1994.
- [83] Yuichi Yamashita and Jun Tani. Emergence of functional hierarchy in a multiple timescale neural network model: a humanoid robot experiment. *PLoS Comput Biol*, Vol. 4, No. 11, p. e1000220, 2008.
- [84] Masaharu Adachi and Kazuyuki Aihara. Associative dynamics in a chaotic neural network. *Neural Networks*, Vol. 10, No. 1, pp. 83–98, 1997.
- [85] Makito Oku and Kazuyuki Aihara. Associative dynamics of color images in a large-scale chaotic neural network. *Nonlinear Theory and Its Applications, IEICE*, Vol. 2, No. 4, pp. 508–521, 2011.
- [86] 岡田昌史, 村上健治. 軌道アトラクタを用いたロボットの同調運動制御によるコミュニケーション原理の表現. *日本ロボット学会誌*, Vol. 25, No. 4, pp. 545–553, 2007.
- [87] Kensuke Ikeda, Kenju Otsuka, and Kenji Matsumoto. Maxwell-bloch turbulence. *Progress of Theoretical Physics Supplement*, Vol. 99, pp. 295–324, 1989.
- [88] Kunihiro Kaneko. Clustering, coding, switching, hierarchical ordering, and control in a network of chaotic elements. *Physica D: Nonlinear Phenomena*, Vol. 41, No. 2, pp. 137–

- 172, 1990.
- [89] Ichiro Tsuda. Chaotic itinerancy as a dynamical basis of hermeneutics in brain and mind. *World Futures: Journal of General Evolution*, Vol. 32, No. 2-3, pp. 167–184, 1991.
- [90] Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural networks. *Physical review letters*, Vol. 61, No. 3, p. 259, 1988.
- [91] Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyong Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, Vol. 16, No. 5, pp. 1190–1208, 1995.
- [92] Auke Jan Ijspeert, Alessandro Crespi, Dimitri Ryczko, and Jean-Marie Cabelguen. From swimming to walking with a salamander robot driven by a spinal cord model. *Science*, Vol. 315, No. 5817, pp. 1416–1420, 2007.
- [93] Silke Steingrube, Marc Timme, Florentin Wörgötter, and Poramate Manoonpong. Self-organized adaptation of a simple neural circuit enables complex robot behaviour. *Nature physics*, Vol. 6, No. 3, p. 224, 2010.
- [94] Chengju Liu, Danwei Wang, and Qijun Chen. Central pattern generator inspired control for adaptive walking of biped robots. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 43, No. 5, pp. 1206–1215, 2013.
- [95] Dai Owaki, Takeshi Kano, Ko Nagasawa, Atsushi Tero, and Akio Ishiguro. Simple robot suggests physical interlimb communication is essential for quadruped walking. *Journal of The Royal Society Interface*, Vol. 10, No. 78, p. 20120669, 2013.
- [96] Jake P Stroud, Mason A Porter, Guillaume Hennequin, and Tim P Vogels. Motor primitives in space and time via targeted gain modulation in cortical networks. *Nature neuroscience*, Vol. 21, No. 12, p. 1774, 2018.
- [97] Matthew G Perich, Juan A Gallego, and Lee E Miller. A neural population mechanism for rapid learning. *Neuron*, Vol. 100, No. 4, pp. 964–976, 2018.
- [98] Tomohiro Taniguchi, Nozomi Akashi, Hirofumi Notsu, Masato Kimura, Hiroshi Tsukahara, and Kohei Nakajima. Chaos in nanomagnet via feedback current. *Physical Review B*, Vol. 100, No. 17, p. 174425, 2019.
- [99] Terufumi Yamaguchi, Nozomi Akashi, Kohei Nakajima, Sumito Tsunegi, Hitoshi Kubota, and Tomohiro Taniguchi. Synchronization and chaos in a spin-torque oscillator with a perpendicularly magnetized free layer. *Physical Review B*, Vol. 100, No. 22, p. 224422, 2019.
- [100] Akira Kamimaki, Tomoyuki Kubota, Sumito Tsunegi, Kohei Nakajima, Tomohiro Taniguchi, Julie Grollier, Vincent Cros, Kay Yakushiji, Akio Fukushima, Shinji Yuasa, et al. Chaos in spin-torque oscillator with feedback circuit. *Physical Review Research*,

- Vol. 3, No. 4, p. 043216, 2021.
- [101] Jacob Torrejon, Mathieu Riou, Flavio Abreu Araujo, Sumito Tsunegi, Guru Khalsa, Damien Querlioz, Paolo Bortolotti, Vincent Cros, Kay Yakushiji, Akio Fukushima, et al. Neuromorphic computing with nanoscale spintronic oscillators. *Nature*, Vol. 547, No. 7664, pp. 428–431, 2017.
- [102] Taishi Furuta, Keisuke Fujii, Kohei Nakajima, Sumito Tsunegi, Hitoshi Kubota, Yoshishige Suzuki, and Shinji Miwa. Macromagnetic simulation for reservoir computing utilizing spin dynamics in magnetic tunnel junctions. *Physical Review Applied*, Vol. 10, No. 3, p. 034063, 2018.
- [103] Sumito Tsunegi, Tomohiro Taniguchi, Kohei Nakajima, Shinji Miwa, Kay Yakushiji, Akio Fukushima, Shinji Yuasa, and Hitoshi Kubota. Physical reservoir computing based on spin torque oscillator with forced synchronization. *Applied Physics Letters*, Vol. 114, No. 16, p. 164101, 2019.
- [104] Nozomi Akashi, Terufumi Yamaguchi, Sumito Tsunegi, Tomohiro Taniguchi, Mitsuhiro Nishida, Ryo Sakurai, Yasumichi Wakao, and Kohei Nakajima. Input-driven bifurcations and information processing capacity in spintronics reservoirs. *Physical Review Research*, Vol. 2, No. 4, p. 043303, 2020.
- [105] Terufumi Yamaguchi, Nozomi Akashi, Kohei Nakajima, Hitoshi Kubota, Sumito Tsunegi, and Tomohiro Taniguchi. Step-like dependence of memory function on pulse width in spintronics reservoir computing. *Scientific reports*, Vol. 10, No. 1, pp. 1–8, 2020.
- [106] Terufumi Yamaguchi, Nozomi Akashi, Sumito Tsunegi, Hitoshi Kubota, Kohei Nakajima, and Tomohiro Taniguchi. Periodic structure of memory function in spintronics reservoir with feedback current. *Physical Review Research*, Vol. 2, No. 2, p. 023389, 2020.
- [107] Ryunosuke Nishimoto and Jun Tani. Development of hierarchical structures for actions and motor imagery: a constructivist view from synthetic neuro-robotics study. *Psychological Research PRPF*, Vol. 73, No. 4, pp. 545–558, 2009.
- [108] Yasuo Kuniyoshi and Shinji Sangawa. Early motor development from partially ordered neural-body dynamics: experiments with a cortico-spinal-musculo-skeletal model. *Biological cybernetics*, Vol. 95, No. 6, p. 589, 2006.
- [109] Stefan L Frank and Henrik Jacobsson. Sentence-processing in echo state networks: a qualitative analysis by finite state machine extraction. *Connection Science*, Vol. 22, No. 2, pp. 135–155, 2010.
- [110] Nikolaos Gianniotis, Sven D Kügler, Peter Tiño, and Kai L Polsterer. Model-coupled autoencoder for time series visualisation. *Neurocomputing*, Vol. 192, pp. 139–146, 2016.
- [111] Hendrik Strobelt, Sebastian Gehrmann, Hanspeter Pfister, and Alexander M Rush. Lstmvis:

-
- A tool for visual analysis of hidden state dynamics in recurrent neural networks. *IEEE transactions on visualization and computer graphics*, Vol. 24, No. 1, pp. 667–676, 2017.
- [112] Kuniyiko Kaneko and Ichiro Tsuda. Chaotic itinerancy. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, Vol. 13, No. 3, pp. 926–936, 2003.
- [113] Hisanori Itoh and Masahide Kimoto. Multiple attractors and chaotic itinerancy in a quasi-geostrophic model with realistic topography: Implications for weather regimes and low-frequency variability. *Journal of the atmospheric sciences*, Vol. 53, No. 15, pp. 2217–2231, 1996.
- [114] Jihoon Park, Hiroki Mori, Yuji Okuyama, and Minoru Asada. Chaotic itinerancy within the coupled dynamics between a physical body and neural oscillator networks. *PloS one*, Vol. 12, No. 8, p. e0182518, 2017.
- [115] Ichiro Tsuda, Edger Koerner, and Hiroshi Shimizu. Memory dynamics in asynchronous neural networks. *Progress of Theoretical Physics*, Vol. 78, No. 1, pp. 51–71, 1987.
- [116] Ahmadreza Ahmadi and Jun Tani. A novel predictive-coding-inspired variational rnn model for online prediction and recognition. *Neural computation*, Vol. 31, No. 11, pp. 2025–2074, 2019.
- [117] Per Bak, Chao Tang, and Kurt Wiesenfeld. Self-organized criticality: An explanation of the 1/f noise. *Physical review letters*, Vol. 59, No. 4, p. 381, 1987.
- [118] Dietmar Plenz, Tiago L Ribeiro, Stephanie R Miller, Patrick A Kells, Ali Vakili, and Elliott L Capek. Self-organized criticality in the brain. *arXiv preprint arXiv:2102.09124*, 2021.
- [119] Ken Caluwaerts, Michiel D’Haene, David Verstraeten, and Benjamin Schrauwen. Locomotion without a brain: physical reservoir computing in tensegrity structures. *Artificial life*, Vol. 19, No. 1, pp. 35–66, 2013.
- [120] Ryo Terajima, Katsuma Inoue, Shogo Yonekura, Kohei Nakajima, and Yasuo Kuniyoshi. Behavioral diversity generated from bodyenvironment interactions in a simulated tensegrity robot. *IEEE Robotics and Automation Letters*, 2021.
- [121] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, Vol. 2, No. 5, pp. 359–366, 1989.
- [122] Moshe Leshno, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, Vol. 6, No. 6, pp. 861–867, 1993.
- [123] Stephen Boyd and Leon Chua. Fading memory and the problem of approximating nonlinear operators with volterra series. *IEEE Transactions on circuits and systems*, Vol. 32, No. 11, pp. 1150–1161, 1985.

- [124] Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural computation*, Vol. 14, No. 11, pp. 2531–2560, 2002.
- [125] Wolfgang Maass and Henry Markram. On the computational power of circuits of spiking neurons. *Journal of computer and system sciences*, Vol. 69, No. 4, pp. 593–616, 2004.
- [126] Lyudmila Grigoryeva and Juan-Pablo Ortega. Echo state networks are universal. *Neural Networks*, Vol. 108, pp. 495–508, 2018.
- [127] Lukas Gonon and Juan-Pablo Ortega. Reservoir computing universality with stochastic inputs. *IEEE transactions on neural networks and learning systems*, Vol. 31, No. 1, pp. 100–112, 2019.
- [128] Lukas Gonon and Juan-Pablo Ortega. Fading memory echo state networks are universal. *Neural Networks*, Vol. 138, pp. 10–13, 2021.
- [129] Ipei Shimada and Tomomasa Nagashima. A numerical approach to ergodic problem of dissipative dynamical systems. *Progress of theoretical physics*, Vol. 61, No. 6, pp. 1605–1616, 1979.
- [130] 池口徹. カオス時系列解析の基礎と応用. 産業図書, 2000.
- [131] Randall D Beer. A dynamical systems perspective on agent-environment interaction. *Artificial intelligence*, Vol. 72, No. 1-2, pp. 173–215, 1995.
- [132] Herbert Jaeger. *Identification of behaviors in an agent's phase space*. Citeseer, 1995. 本研究はテクニカルレポートであり正式には出版されていない。文章は下記の URL においてダウンロードできる。<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.40.3355&rep=rep1&type=pdf>.
- [133] Ichiro Tsuda. Toward an interpretation of dynamic neural activity in terms of chaotic dynamical systems. *Behavioral and Brain Sciences*, Vol. 24, No. 5, pp. 793–810, 2001.
- [134] Ichiro Tsuda. Chaotic itinerancy and its roles in cognitive neurodynamics. *Current opinion in neurobiology*, Vol. 31, pp. 67–71, 2015.
- [135] Christine A Skarda and Walter J Freeman. How brains make chaos in order to make sense of the world. *Behavioral and brain sciences*, Vol. 10, No. 2, pp. 161–173, 1987.
- [136] Charles M Gray, Andreas K Engel, Peter König, and Wolf Singer. Synchronization of oscillatory neuronal responses in cat striate cortex: temporal properties. *Visual neuroscience*, Vol. 8, No. 4, pp. 337–347, 1992.
- [137] Jun Tani and Naohiro Fukumura. Embedding a grammatical description in deterministic chaos: an experiment in recurrent neural learning. *Biological Cybernetics*, Vol. 72, No. 4, pp. 365–370, 1995.
- [138] Jun Tani and Stefano Nolfi. Learning to perceive the world as articulated: an approach

-
- for hierarchical learning in sensory-motor systems. *Neural Networks*, Vol. 12, No. 7-8, pp. 1131–1141, 1999.
- [139] Tomoki Kurikawa and Kunihiko Kaneko. Embedding responses in spontaneous neural activity shaped through sequential learning. *PLoS computational biology*, Vol. 9, No. 3, p. e1002943, 2013.
- [140] Shigetoshi Nara and Peter Davis. Chaotic wandering and search in a cycle-memory neural network. *Progress of Theoretical Physics*, Vol. 88, No. 5, pp. 845–855, 1992.
- [141] Yasuo Kuniyoshi and Shinsuke Suzuki. Dynamic emergence and adaptation of behavior through embodiment as coupled chaotic field. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, Vol. 2, pp. 2042–2049. IEEE, 2004.
- [142] Takashi Ikegami. Simulating active perception and mental imagery with embodied chaotic itinerancy. *Journal of Consciousness Studies*, Vol. 14, No. 7, pp. 111–125, 2007.
- [143] Hiromichi Suetani. Multiple pattern generations and chaotic itinerant dynamics in reservoir computing. In *International Conference on Artificial Neural Networks*, pp. 76–81. Springer, 2019.
- [144] David Sussillo and LF Abbott. Transferring learning from external to internal weights in echo-state networks with sparse connectivity. *PloS one*, Vol. 7, No. 5, p. e37372, 2012.
- [145] Gilles Wainrib and Mathieu N Galtier. A local echo state property through the largest lyapunov exponent. *Neural Networks*, Vol. 76, pp. 39–45, 2016.
- [146] Herbert Jaeger, Mantas Lukoševičius, Dan Popovici, and Udo Siewert. Optimization and applications of echo state networks with leaky-integrator neurons. *Neural networks*, Vol. 20, No. 3, pp. 335–352, 2007.
- [147] Peter Davis. Application of optical chaos to temporal pattern search in a nonlinear optical resonator. *Japanese journal of applied physics*, Vol. 29, No. 7A, p. L1238, 1990.
- [148] Yuhei Mori, Peter Davis, and Shigetoshi Nara. Pattern retrieval in an asymmetric neural network with embedded limit cycles. *Journal of Physics A: Mathematical and General*, Vol. 22, No. 11, p. L525, 1989.
- [149] Klaus Nützel, Jenny Kien, Karin Bauer, Jennifer S Altman, and Uwe Krey. Dynamics of diluted attractor neural networks with delays. *Biological cybernetics*, Vol. 70, No. 6, pp. 553–561, 1994.
- [150] Viola Folli, Giorgio Gosti, Marco Leonetti, and Giancarlo Ruocco. Effect of dilution in asymmetric recurrent neural networks. *Neural Networks*, Vol. 104, pp. 50–59, 2018.
- [151] Nils Bertschinger and Thomas Natschläger. Real-time computation at the edge of chaos in recurrent neural networks. *Neural computation*, Vol. 16, No. 7, pp. 1413–1436, 2004.

- [152] Joni Dambre, David Verstraeten, Benjamin Schrauwen, and Serge Massar. Information processing capacity of dynamical systems. *Scientific reports*, Vol. 2, p. 514, 2012.
- [153] Eduardo Izquierdo-Torres and Inman Harvey. Hebbian learning using fixed weight evolved dynamical neural networks. In *2007 IEEE Symposium on Artificial Life*, pp. 394–401. IEEE, 2007.
- [154] Donald Olding Hebb. *The organisation of behaviour: a neuropsychological theory*. Science Editions New York, 1949.
- [155] Christian Klos, Yaroslav Felipe Kalle Kossio, Sven Goedeke, Aditya Gilra, and Raoul-Martin Memmesheimer. Dynamical learning of dynamics. *Physical Review Letters*, Vol. 125, No. 8, p. 088103, 2020.
- [156] Inman Harvey, Ezequiel Di Paolo, Rachel Wood, Matt Quinn, and Elio Tuci. Evolutionary robotics: A new scientific tool for studying cognition. *Artificial life*, Vol. 11, No. 1-2, pp. 79–98, 2005.
- [157] Menachem Stern, Chukwunonso Arinze, Leron Perez, Stephanie E Palmer, and Arvind Murugan. Supervised learning through physical changes in a mechanical system. *Proceedings of the National Academy of Sciences*, Vol. 117, No. 26, pp. 14843–14850, 2020.
- [158] Menachem Stern, Daniel Hexner, Jason W Rocks, and Andrea J Liu. Supervised learning in physical networks: From machine learning to learning machines. *Physical Review X*, Vol. 11, No. 2, p. 021045, 2021.
- [159] Beren Millidge, Alexander Tschantz, and Christopher L Buckley. Predictive coding approximates backprop along arbitrary computation graphs. *arXiv preprint arXiv:2006.04182*, 2020.
- [160] Esteban Real, Chen Liang, David So, and Quoc Le. Automl-zero: Evolving machine learning algorithms from scratch. In *International Conference on Machine Learning*, pp. 8007–8019. PMLR, 2020.
- [161] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning —a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, Vol. 41, No. 9, pp. 2251–2265, 2018.
- [162] Guillaume Bellec, Franz Scherr, Elias Hajek, Darjan Salaj, Robert Legenstein, and Wolfgang Maass. Biologically inspired alternatives to backpropagation through time for learning in recurrent neural nets. *arXiv preprint arXiv:1901.09049*, 2019.
- [163] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. *Neurocomputing*, Vol. 70, No. 1-3, pp. 489–501, 2006.
- [164] Amir F Atiya and Alexander G Parlos. New results on recurrent network training: unifying the algorithms and accelerating convergence. *IEEE transactions on neural networks*,

-
- Vol. 11, No. 3, pp. 697–709, 2000.
- [165] Claudio Gallicchio. Chasing the echo state property. *arXiv preprint arXiv:1811.10892*, 2018.
- [166] Yutaka Yamaguti and Ichiro Tsuda. Functional differentiations in evolutionary reservoir computing networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, Vol. 31, No. 1, p. 013137, 2021.
- [167] Michael C Mackey and Leon Glass. Oscillation and chaos in physiological control systems. *Science*, Vol. 197, No. 4300, pp. 287–289, 1977.
- [168] J Doyne Farmer. Chaotic attractors of an infinite-dimensional dynamical system. *Physica D: Nonlinear Phenomena*, Vol. 4, No. 3, pp. 366–393, 1982.
- [169] Shuji Ishihara and Kunihiro Kaneko. Magic number 7 ± 2 in networks of threshold dynamics. *Physical review letters*, Vol. 94, No. 5, p. 058102, 2005.
- [170] Yichen Shen, Nicholas C Harris, Scott Skirlo, Mihika Prabhu, Tom Baehr-Jones, Michael Hochberg, Xin Sun, Shijie Zhao, Hugo Larochelle, Dirk Englund, et al. Deep learning with coherent nanophotonic circuits. *Nature Photonics*, Vol. 11, No. 7, pp. 441–446, 2017.
- [171] Shahab Torkamani and Eric A Butcher. Delay, state, and parameter estimation in chaotic and hyperchaotic delayed systems with uncertainty and time-varying delay. *International Journal of Dynamics and Control*, Vol. 1, No. 2, pp. 135–163, 2013.