Walter Benjamin Fellowship Final Report

1 General information

DFG reference number: QU 579/1-1
Project number: 493789610
Project title: Validated computation of patterns in recurrent neural networks
Name of the applicant: Elena Queirolo
Official address: IRMAR, Université de Rennes, Campus de Beaulieu, Rennes (FR)
Name of the host individual and institution: Christian Kuehn, TUM
Reporting period (entire funding period): 02.2022 - 09.2024

2 Summary

Machine Learning (ML), often referred to as Artificial Intelligence (AI), has rapidly expanded into various applications in recent years, from self-driving cars to image generation tools and the widespread use of ChatGPT. While these advancements are impressive, they also raise the question: How do these systems work? Although this is a complex and ongoing question, a complete answer remains elusive.

At a basic level, a typical ML model consists of two main components: the architecture (or structure) and the parameters (or weights). The architecture is set in advance, while the parameters are initially random and adjusted during a process called *learning*. The model is trained on data, gradually improving by optimising its parameters to achieve a desired level of accuracy.

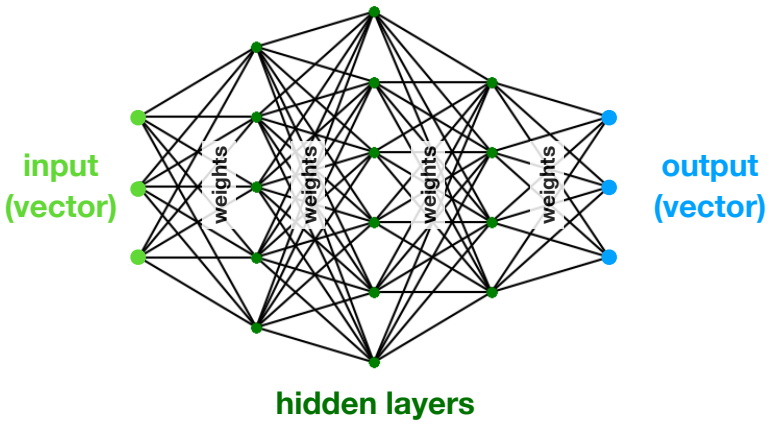

Figure 1: a representation of a Neural Network. In this project, all layers are similar, but in general each layer includes weights and non-linear functions applied to each element.

The trained weights are central to how the ML model functions, but they result from a complex optimization process influenced by the model's architecture, the data used, and the optimization technique itself. Given the vast number of parameters and data involved, fully understanding or monitoring how these models work can be challenging.

This project is part of the broader field of *explainable AI* (XAI), which seeks to improve our understanding of AI models. Our approach draws on concepts from dynamical systems, a branch of mathematics that studies processes that evolve over time. We view an ML model as a dynamical system that maps inputs to outputs, and we apply dynamical systems theory to explore its behaviour.

One key concept we focused on is *periodic orbits*, where a system returns to its starting state after a certain period. In dynamical systems, periodic orbits are important because they often signal the onset of chaotic behaviour. In ML, periodic orbits can indicate problems in the learning process—such as when an algorithm fails to learn meaningful patterns and instead gets stuck in repetitive cycles.

Our first main finding was the detection of *Hopf bifurcations*, which mark the onset of periodic orbits in ML models. This provides a way to identify when a model's learning process is stagnating.

We then turned our attention to detecting chaotic behaviour in trained ML models. This area of research is ongoing, but we have already made progress in developing methods for identifying chaos in these systems.

In summary, our work contributes to efforts to better understand and interpret ML models, with a particular focus on identifying periodic and chaotic dynamics that could indicate issues with learning. While more research is needed, our findings offer a valuable step toward making machine learning systems more transparent and comprehensible.

3 Scientific progress report

Background and objectives of the project

The methodology of this project aims to bridge machine learning algorithms with tools from dynamical systems. A significant portion of the research is dedicated to developing this connection and identifying dynamical system techniques that can be applied to machine learning models.

With this toolkit in hand, our long-term goal is to describe the dynamics learned by AI models, particularly neural networks. We are interested in identifying and classifying behavioural patterns, as well as understanding how chaos emerges in trained models. Additionally, we aim to explore how the architecture and training algorithms influence these behaviours.

This project is situated within the growing field of "explainable AI," which seeks to demystify AI systems and make them more understandable to human users. The field has two main components: “interpretability” (how a model works) and “explainability” (why a model was trained in a particular way). Our initial results primarily focus on interpretability, with plans to extend into explainability and the development of control mechanisms that can guide the training phase toward more meaningful outcomes.

On a more practical level, one architecture that closely aligns with dynamical systems concepts is the neuralODE [1]. This architecture refines the ResNet (Residual Network) model and is designed

to process time-dependent inputs and produce time-dependent outputs. What sets it apart is its flexibility—both the input and output lengths are dynamic and determined at runtime, rather than being fixed by the architecture itself.

Practically, every step of the algorithm has a history (stored in the vector) and it updates it following the formula , where are parameters (weights and biases) learned during training. With some additional restrictions, it can be stated that this approach is a discretisation through Euler method of an ordinary differential equation (ODE) for a new function strongly related to the initial and for some parameters

In practice, every step of a neuralODE algorithm has a history, which is stored in the vector $\mathbf{h}_t$ and updated according to the formula $\mathbf{h}_{t+1} = \mathbf{h}_t + f(\mathbf{h}_t, \theta_t)$. The weights θ_t that govern this update are learned during training. Under certain conditions, this approach can be seen as a discretisation of the ordinary differential equation (ODE) $\frac{d}{dt}\mathbf{h}(t) = F(\mathbf{h}(t), \theta)$ using the Euler method, where F is a new function related to the initial f, with parameters θ that may be time-dependent (though typically assumed to be constant). This ODE formulation allows us to view the inner workings of a neuralODE as a dynamical system, aligning with our goal.

A challenge when applying traditional dynamical systems concepts to AI algorithms is the sheer scale at which neural networks operate. On one hand, the phase space (the space in which the dynamics occur) expands from a handful of dimensions in classical dynamics to hundreds in AI. On the other, the number of parameters—usually just one or two in classical analysis—grows to thousands of weights and biases, learned during training. As a result, traditional pen-and-paper approaches quickly become impractical for such large-scale problems.

To address this, we turn to *validated numerics*, a field focused on using computers to return mathematically rigorous results. The core idea is that computers can only compute approximate solutions, as all numbers in a computer must be represented by a finite sequence of bits. Therefore, asking a computer to solve an equality between two non-trivial numbers is mathematically impossible—approximations will always introduce small errors. However, one operation that remains mathematically rigorous is comparison (i.e., < or >). By leveraging this, we can enclose the exact solution within computable upper and lower bounds. This means that, whenever a computer runs a specific operation, it provides rigorous bounds on the exact (but unknown) result [2]. When combined with techniques like *radii polynomials* [4], this approach allows us to derive mathematically proven results in dynamical systems [3].

By merging machine learning, dynamical systems theory, and validated numerics, our aim is to rigorously understand the behaviour of a neuralODE. Our primary focus is on identifying and analysing behavioural patterns, such as periodic orbits, and the early signs of chaos, including the computation of Lyapunov exponents.

Description of the project-specific results and findings

The first technique we apply in this context is the search and validation of *Hopf bifurcations* [5]. In this paper, we demonstrate how viewing neuralODEs through the lens of dynamical systems provides valuable insights into the dynamical behaviour of AI models—both before and after training, and with or without additional structural modifications. We achieve this by first interfacing the neuralODE with straightforward numerical methods, such as a Newton map for detecting algebraic Hopf bifurcations, followed by validated numerics to confirm the results. This combination allows for a deeper understanding, enabling us to distinguish between solutions that may appear numerically similar but are shown to differ during the validation process.

Detecting Hopf bifurcations in a one-parameter problem allows us to identify the onset of periodic orbits in the system. These orbits can hinder the learning process, but fortunately, in most of the examples we examined, they were found to be unstable—meaning that nearby orbits would diverge from them. Additionally, these periodic orbits tend to exist only within narrow parameter intervals. Therefore, a careful analysis of the trained parameters can help us quickly determine whether the training has led to a region where periodic orbits are present.

While this published result is directly related to the core of our project, two other papers have emerged from ongoing collaborations that were significantly influenced by this central research.

In paper [6], validated numerics techniques, typically applied to one-parameter families of solutions, have been extended to two-parameter families, demonstrating that high-dimensional parameter validation is achievable. This has clear relevance to AI research, where the parameter space is often high-dimensional.

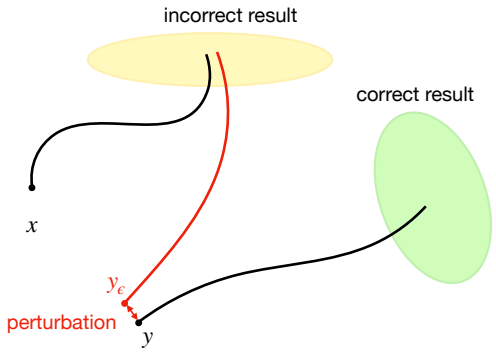

Figure 2: representation of an adversarial attack as two orbits diverging

The latest preprint [7] introduces numerical methods developed for studying biological network dynamics. The key contribution for our project is the ability to explore high-dimensional parameter spaces while extracting valuable insights into the global dynamics of the system. This collaboration was also the springboard to the organisation of the “Computing combinatorial dynamics in high dimensional biological networks” workshop in fall 2023 at the Lorenz Center (Leiden, NL). The main focus of this work is efficient approaches to the equilibrium search problem in large phase spaces—a challenge that was initially overlooked in paper [5], where a structure was chosen that allowed for a unique equilibrium. However, this issue would resurface with different parameter structures. Therefore, the results presented in this preprint, although in a different context, are relevant to the ongoing development of our research.

Finally, paper [8] emerged from an impromptu collaboration within the hosting group, highlighting the vitality and interconnectivity of the research team. In this paper, validated numerics are used to support and extend analytical results in forced systems. While this work is more distant from the core of our project, it demonstrates how validated numerics is gaining recognition within the

Figure 3: a representation of the learned dataset, called the “moon problem”, where colours indicate the two categories that the algorithm should learn.

broader mathematical community as a valuable tool for providing rigorous proof to numerical computations, which aligns with our goal in uncovering the dynamics of AI systems.
In addition to the published results within the project's timeframe, several lines of research are still ongoing. Three key areas of investigation remain open and are actively being explored:
-detection of chaos in AI through Lyapunov exponents,
-use of computer validation of Lyapunov exponents in AI,
-slow-fast interpretation of the neuralODE dynamics.

Detection of chaos in AI through Lyapunov exponents

Consider a trained neuralODE as a flow within a specific region of phase space. It is hypothesised that this flow exhibits notable properties, particularly near decision boundaries (or separatrices of the dynamical system), due to the behaviour of adversarial attacks. Adversarial attacks involve perturbations applied to correctly processed inputs in such a way that the modified input is misclassified. Famous examples include imperceptible image alterations that deceive classifiers or objects in photos that override the initial categorisation. These behaviours are well-documented and ubiquitous [9], making it crucial to understand their origins in order to build more trustworthy AI models.
In the context of a neuralODE, an adversarial result can be visualised as two orbits that start close together but diverge towards different end positions, as shown in Figure 2. If we assume that for nearly any perturbation size and all initial conditions, a perturbation exists that leads to this effect, we can picture a dynamic where "natural" (unperturbed) inputs belong to a manifold of well-behaved dynamics but are consistently near chaotic regions. It is also known that larger perturbations can often be chosen to force a specific incorrect output. Thus, in the vicinity of any given natural input, initial conditions leading to all possible outputs are represented. This dynamical behaviour closely resembles chaos, where mixing properties ensure that every neighbourhood is mapped to all potential outcomes.
To investigate and validate this hypothesis, we are collaborating with Tobias Woehrer and Christian Kuehn to use *Lyapunov exponents* (LE) as a tool for detecting both decision boundaries and chaotic regions in trained networks. This approach aims to provide deeper insights into the dynamics underlying adversarial vulnerabilities in neuralODEs.
Lyapunov exponents are coefficients that measure the average divergence between two neighbouring orbits for time tending to infinity. In our case, we focus on Finite Time LE (FTLE), defined as $\frac{1}{2t}\log(Y(t)Y^T(t))$, where $Y(t)$ solves the ODE $\dot{Y}(t) = J(f,x)Y$ where $J(f,x)$ denotes

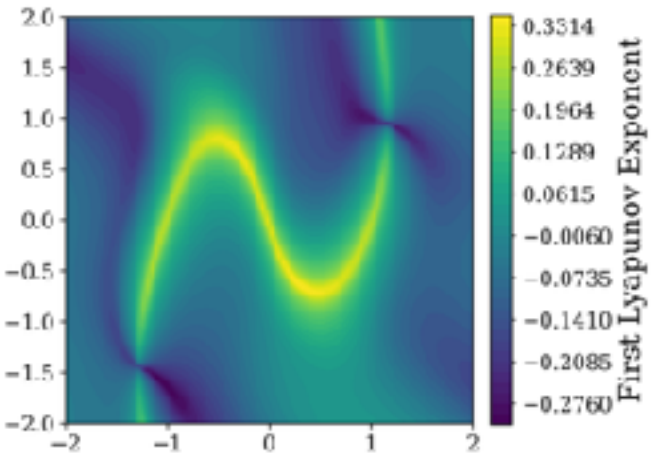

Figure 4: the computed FTLE w.r.t. the position.

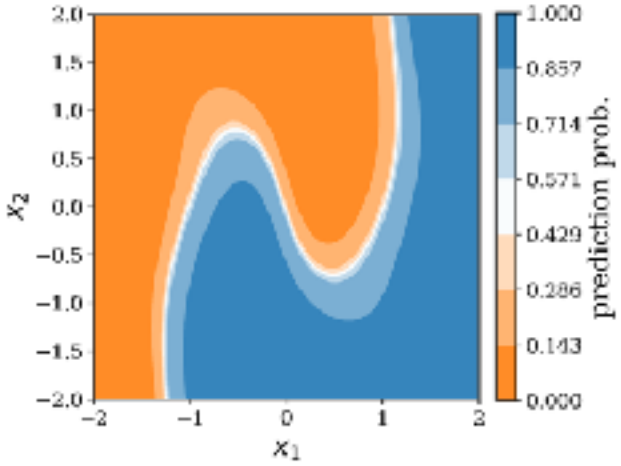

Figure 5: the learned probability distribution indicating categorisation w.r.t. position.

the Jacobian of f w.r.t. x. The usual initial condition considered is $Y(0) = Id$, but any initial value of full range in known to converge to the same values.

Initial results are positive, as shown in Figures 4, but still exhibit significant numerical instability. In Figure 5 we see how the Lyapunov exponents give insight on the decision boundary, while still exhibiting additional, not yet understood, oscillations. Incidentally, the decision boundary is efficiently computable just in the simplest of problems, thus LE would give a wider applicable method of detecting them. Previous works, such as [10], which use Lyapunov exponents (LE) to rank the stability of inputs, and [11], where LE are applied to detect counterfeit inputs, have explored similar concepts. However, both papers lack the mathematical rigour necessary for reliable LE predictions. While the algorithms used to compute LE approximations are known to converge as time approaches infinity, they exhibit poor short-term convergence. Therefore, applying these methods to short time scales, such as those typically used in machine learning, is not mathematically sound. Addressing this limitation is the challenge we aim to overcome in the coming months.

Use of computer validation of Lyapunov exponents in AI

One approach we are exploring to ensure the stability of numerical computations related to Lyapunov exponents is the use of validated numerics. This method, already introduced in this project to verify the mathematical reliability of numerical results, appears well-suited for our needs. However, a key limitation arises: in standard implementations, applying validation to a given computation typically increases its computational cost by a factor of four at best, and potentially much more if further refinement is required. When interfacing with AI applications, which are themselves large and computationally demanding, this presents a practical challenge in creating an algorithm that is generically applicable to large dynamical systems. To address this, we are not only using existing software, such as CAPD [12], but also adapting both the problem and the method to extend the applicability of validated approaches. This naturally leads into our next area of research: the slow-fast decomposition of neural network dynamics.

Slow-fast interpretation of the neuralODE dynamics

In the previous sections, we discussed how the high dimensionality of AI applications presents a bottleneck for implementing our ideas, as these systems often scale poorly with increasing dimensions. A standard dimension reduction approach is unlikely to offer a solution, as such methods are typically tailored to case-specific analyses, which contrasts with our goal. Moreover, these techniques typically reduce dimensions one at a time, which is effective in low-dimensional settings but insufficient when dealing with problems involving hundreds of dimensions. As a result, a different approach is needed. Our strategy is to draw inspiration from slow-fast dynamics. In this framework, a system of the form $\dot{x} = f(x)$, for $x \in \mathbb{R}^n, n \geq 2$, is decomposed into slow components x_s and fast components x_f, where $x = (x_s, x_f)$ and we can write

$$\begin{cases} \dot{x}_s = \epsilon f_s(x_s, x_f), \\ \dot{x}_f = f_f(x_s, x_f), \end{cases}$$

for some ϵ small. The slow component is expected to evolve on a different time scale than the fast component. Depending on the time scale of interest, it may be reasonable to treat the slow component as nearly constant, or alternatively, to consider it changing so rapidly that it effectively behaves like noise, in which case it can be studied from a purely statistical perspective. When this separation of time scales is feasible, it is referred to as *scale separation*.

In our neuralODE applications, the system we usually consider is of the form

$$\dot{x} = \tanh(Wx + b)$$

where W and b are the trained weights and biases. In this case, we are interested in generating an algorithm that extracts from the system a scaled separated version of it of the form

$$\begin{cases} \dot{x}_s = \tanh(W_{ss}x_s + W_{fs}x_f + b_s), \\ \dot{x}_f = \tanh(W_{sf}x_s + W_{ff}x_f + b_f), \end{cases}$$

where we chose the coordinate system in such a way that the first equation's right hand side is on a smaller scale than the second equation's one.

Currently, two key research areas are being explored. First, we are focused on determining where scale separation occurs. Second, we aim to identify which terms are most relevant in applications, given the interplay between the typical time scale of the applications and the scaling between slow and fast components. Specifically, the question is whether we should treat the slow terms as constant or the fast terms as random.

Deviations from the original concept

The original project proposed three main research areas in the study of dynamics in AI: the detection of patterns (P1), the determination of their stability (P2), and the computation of a bifurcation diagram (P3).

The initial concept of the project was to treat machine learning behaviour as a dynamical system on a graph. However, this assumption proved to be cumbersome for two interrelated reasons. First, tracking the behaviour of all internal nodes in a given AI model is computationally intensive, given the large number of nodes involved. Second, since neural networks are directed graphs, focusing on the hidden layers allows us to leverage the network's dynamics interpretation. However, the output of the network is a direct result of the hidden layers' dynamics and is typically lower-dimensional. Thus, studying the output instead of the hidden layers would yield the same information while requiring less computational effort. As a result, a key shift in the approach lies in how the studied system is interpreted dynamically.
It quickly became apparent that detecting differences between patterns of behavior on the short timescales typical in AI applications posed a challenge. To address this, it was necessary to first determine the stability of orbits in order to understand their relative behavior, which was essential for classification. As a result, the order of the project's components was revised, prioritizing stability over pattern detection and enabling the computation of (partial) bifurcation diagrams earlier than originally planned.
Thus, partial progress on P1, P2, and P3 for periodic orbits has been achieved in the first publication [5]. Stability of validated patterns was also addressed in [8], where Lyapunov exponents were successfully computed and validated for two branches of periodic orbits.
For similar reasons, the detection of chaos, initially considered a stretch goal, has become an active research focus in the final year of the project, with a related publication expected in early 2025.

Handling of research data and data infrastructures

Not applicable: the project did not generate any data requiring handling or storage.

Bibliography

[1] Chen, Ricky TQ, et al. "Neural ordinary differential equations." *Advances in neural information processing systems* 31 (2018).
[2] Rump, Siegfried M. "INTLAB—interval laboratory." *Developments in reliable computing*. Dordrecht: Springer Netherlands, 1999. 77-104.
[3] Zgliczyński, Piotr, and Marian Gidea. "Covering relations for multidimensional dynamical systems." *Journal of Differential Equations* 202.1 (2004): 32-58.
[4] Hungria, Allan, Jean-Philippe Lessard, and J. Mireles James. "Rigorous numerics for analytic solutions of differential equations: the radii polynomial approach." *Mathematics of Computation* 85.299 (2016): 1427-1459.
[5] Queirolo, Elena, and Kuehn, Christian "Computer Validation of Neural Network Dynamics: A First Case Study.", Discrete and Continuous Dynamical Systems - Series B, doi:10.3934/dcdsb.2024145

[6] Church, Kevin, and Elena Queirolo. "Computer-assisted proofs of Hopf bubbles and degenerate Hopf bifurcations." *Journal of Dynamics and Differential Equations* (2023): 1-55
[7] Kepley, Shane, Konstantin Mischaikow, and Elena Queirolo. "Global analysis of regulatory network dynamics: equilibria and saddle-node bifurcations." *arXiv preprint arXiv:2204.13739* (2024).
[8] Longo, Iacopo, and Queirolo, Elena and Kuehn, Christian “On the transition between autonomous and nonautonomous systems: the case of FitzHugh-Nagumo's model”, Chaos: An Interdisciplinary Journal of Nonlinear Science, doi??
[9] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *arXiv preprint arXiv:1412.6572* (2014).
[10] Storm, L., et al. "Finite-time Lyapunov exponents of deep neural networks." *Physical Review Letters* 132.5 (2024): 057301.
[11] Pedraza, Anibal, Oscar Deniz, and Gloria Bueno. "Lyapunov stability for detecting adversarial image examples." *Chaos, Solitons & Fractals* 155 (2022): 111745.
[12] Kapela, Tomasz, et al. "CAPD:: DynSys: a flexible C++ toolbox for rigorous numerical analysis of dynamical systems." *Communications in nonlinear science and numerical simulation* 101 (2021): 105578.

4 Published Project Results

4.1 Publications with scientific quality assurance

Queirolo, Elena, and Kuehn, Christian "Computer Validation of Neural Network Dynamics: A First Case Study.”, Discrete and Continuous Dynamical Systems - Series B, doi:10.3934/dcdsb.2024145

Longo, Iacopo, and Queirolo, Elena and Kuehn, Christian “On the transition between autonomous and nonautonomous systems: the case of FitzHugh-Nagumo's model”, Chaos: An Interdisciplinary Journal of Nonlinear Science, https://doi.org/10.1063/5.0234833

Church, Kevin, and Elena Queirolo. "Computer-assisted proofs of Hopf bubbles and degenerate Hopf bifurcations." *Journal of Dynamics and Differential Equations* (2023): 1-55.

4.2 Other publications and published results

Kepley, Shane, Konstantin Mischaikow, and Elena Queirolo. "Global analysis of regulatory network dynamics: equilibria and saddle-node bifurcations." *arXiv preprint arXiv:2204.13739* (2024).

4.3 Patents (applied for and granted)

Not applicable.