THE UNIVERSITY OF CHICAGO

SCULPTING FOLDING PATHWAYS THROUGH PHYSICAL LEARNING IN MECHANICAL SYSTEMS

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF PHYSICS

BY
CHUKWUNONSO ARINZE

CHICAGO, ILLINOIS
AUGUST 2022

This work is dedicated to the memory of my grandmother, Manari Felicia Unoma Okolo and my uncle, Azubike Okolo, who unfortunately did not live long enough to see me graduate.

There are things which seem incredible to most men who have not studied Mathematics - Archimedes

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

First and foremost I am extremely grateful to my supervisors, Prof. Arvind Murugan and Prof. Sid Nagel for their invaluable advice, continuous support, and patience during my PhD study. Their immense knowledge and experience have encouraged me in all the time of my academic research and daily life. Prof. Murugan, my primary adviser, taught me patience, empathy and kindness. I am blessed to have worked with him. I would also like to thank Prof. Littlewood, Prof. Palmer and Prof. Kasthuri for their technical support and advice in the course of my graduate study. I am indebted to Dr. Stern who collaborated with me on all my research work. I am thankful for the patience, kindness and generosity he showed me in the course of our working together.

I would like to thank the members in the Murugan lab - Kabir, Vedant, Martin, and Weerapat. I would also like to thank Dr. Reid and Amy Schulz for all their help especially with administrative compliance and checking on my well being all throughout my years of study. It is their kind help and support that have made my study and life as an international student a wonderful time.

Finally, I would like to express my gratitude to my parents (Ogochukwu and Chinweze), siblings (Chineme, Okeoma, and Ifunanya), and startup co-founders (Princess Oti, Obinna Agba, Akintunde Ajayi and Bello Atolagbe) my girlfriend, Chidimma Nwabueze. Without their tremendous love and encouragement in the past few years, it would be impossible for me to complete my study.

ABSTRACT

Creased sheets can possess an exponential number of folding pathways accessible from the flat state. This property poses an engineering challenge in the design of specialized self-folding origami patterns with one or a few prescribed folding pathways. Hence, an alternative means to eliminating undesired folding pathways via physical training is sought. Creased sheets are folded along desired folding pathways and their initially equal crease stiffnesses are allowed to dynamically evolve according to a local rule dependent on folding strain. It is found that undesired folding pathways are eliminated in saddle-node bifurcations, leaving behind only the desired folded pathway. Thus, physical folding, combined with a plasticity rule for crease stiffness, can naturally arrive at design parameters needed for non-linear behaviors that are hard to predict otherwise.

This work also explores the capability of mechanical networks to accomplish more complex tasks, like classification of forces. It demonstrates supervised learning in self-folding origami patterns. Self folding origamis are able to learn from examples through a simple dynamic physical learning rule. These self-folding origamis were able to identify the different classes of the iris flowers based on their features, such as the length and width of the petals and sepals of the flower.

CHAPTER 1

INTRODUCTION

One of the characteristics of intelligence is the ability to learn. Learning is broadly defined as the ability of a system to gain new capabilities from experiencing examples of these capabilities. Learning theory is often studied in the context of "intelligent" systems, be they biological neural networks or non-biological artificial neural networks [91, 59] While the mechanistic details of systems in question can differ greatly, there are several aspects common to any 'learning' process.

Take a student in a typical course for example. Students learn from seeing examples of course materials worked out by a teacher on a blackboard. They take down notes which go over worked out example problems. Most textbooks teach a course by working out example problems which serve to guide students on their ability to solve problems which they have not seen or encountered. Students then go on to test their understanding and newly gained knowledge by attempting to solve a new problem set which they had hitherto not seen or experienced. The students are graded for their performance on this test problem set. Based on their grade on this test problem set, students can evaluate the extent of their newly gained knowledge, or how well they have learnt a new skill. A low score would require a student to return back to the course material and go over worked examples attempting to improve their understanding and knowledge of the course material. After revising over the course material with a teacher or textbook, they may re-attempt a new set of test problems, to evaluate their understanding. This cycle of tests on new problem sets and revisions over older problem sets continue until a pass grade is attained. At that point, the student is judged to have gained good knowledge of the skill set being learned. This analogy with a student in a classroom highlights the key elements of learning:

• First, a teacher or textbook exists to guide the student through a worked-out example problem.

• Second, students previously had no knowledge or capability to solve this problem and only gained understanding of solving this problem by following through an example problem and adjusting their initial prejudices, biases, and knowledge. They must be willing to accept new concepts and ideas as they follow through with the example problem.

• Third, the students test their new understanding/knowledge by attempting to solve new problem sets which they have not attempted to solve before.

• Fourth, based on their performance on new problem sets, hitherto seen before, students' knowledge of the new skill to be learnt is evaluated

• Fifth, the student cycles back and forth between revising over old problem sets and doing new problem sets until a set pass grade is exceeded.

However, while the above analogy has been demonstrated in animate systems like a student, these same five highlighted themes hallmark learning in inanimate systems like machines. Machine learning algorithms are characterized by learning from examples called training data sets (first theme) and evaluating the efficacy of the learning algorithms with a new data set called the test data set (third theme). A criteria/parameter called a cost function which measures the performance of the machine learning algorithm on the test data set is established (fourth theme). This cost function is typically minimized. Unlike the student whose performance is evaluated by a high grade, here, the learning algorithm is deemed well trained by a low value of its cost function. Thus, the process of learning in machines involves the minimization of the cost function. To minimize the cost function, parameters of the machine learning algorithm are adjusted during the training session when the machine learns with examples from the training data set (second theme). Just as it is with the student, the algorithm is cycled between rounds of training and testing till the cost function is minimized below the set target and the algorithm is deemed to have been optimally trained.

With the similarities that exist between how animate biological systems like a student in

a classroom learns and an inanimate system like a computer trained for facial recognition, it seems plausible to wonder if there are universal principles that govern learning irrespective of whatever type of systems – biological, physical, or chemical – are being trained.

Moreover, of the five highlighted themes, none required the existence of a centralized "brain" device, that is, a sophisticated computational processing unit to make decisions based on some defined program. To take it further, if there are universal themes that define the process of learning, does intelligence, the ability to learn, require the need for "brain power", a sophisticated computational processing unit like a human brain or the central processing unit of a computer?

Can low complexity de-centralized physical systems be capable of performing these five highlighted themes learn from examples? The answer is yes! Inanimate "simple" systems without "brain power" are indeed capable of learning new capabilities and skill sets following the five themes earlier highlighted. In particular, disordered mechanical and electrical systems like disordered elastic networks, disordered network of electrical resistors and disordered self-folding origamis have been found to possess properties that enable them to acquire new capabilities via the five themes of learning earlier highlighted[81, 38, 89, 110]. Disordered networks/systems have the unique advantage of a plethora of unconstrained and independent parameters which can be made to evolve during a physical learning session. The disordered nature of these systems allows each parameter to affect the output of the system without much constrain, as compared to ordered systems restrained by various symmetries in the system.

The possibility of disordered mechanical systems to acquire new skills learned from example instances of the desired functionality suggests a paradigm shift in the engineering of new functionalities in materials. Typically, when engineers create a material for a new functionality, they do this from a first-principle engineering point of view. Engineers calculate the parameters of various material properties and the combinations of these material

properties needed to accomplish a specific functionality. They often account for all scenarios in their functionality design, with a focus on the limiting or "boundary" cases. By designing for these "boundary" scenarios/cases, engineers attempt to accommodate for every instance of the desired functionality they hope to accomplish in a material.

After identifying the design constraints, limiting cases, and functionality specifications, engineers deploy first-principle models based on the physics and chemistry of the material that they are working with. These first-principle models often involve computationally intensive optimization problems, as the protocol is to search for material parameter values which satisfy various engineering constrains that encode the desired material functionality. This approach to engineering materials for new functionalities is what is termed "Material Design." While this approach results in very precise material functionality, it has the downside of being computationally intensive and resulting in a static design for theoretically specified use cases.

On the other hand, the possibility of pristine materials acquiring new material functionalities learned from example instances of these functionalities presents a new paradigm for material engineering. Note that the material need not be engineered for every instance of the desired functionality (limiting/" boundary" cases) but rather from examples of the desired functionality. Lastly, there are no first principle computational models with this approach. Therefore, from a computational resource point of view, engineering a material with this approach is computationally cheap. This approach to engineering materials is the "physical learning" approach to material science. However, this engineering approach has several pitfalls. Engineering material functionality from a physical learning approach does not always result in materials with high precision functionalities. Another limitation is that the material in question must be capable of evolving its properties during the course of a training section, in which it experiences instance of the material functionality it intends to learn. These restrictions therefore limit the number of materials that could possibly learn

new functionalities.

Another important difference between the material learning approach and the material design approach is the concept of locality. In the physical learning approach of material science, the untrained materials are subjected to instances of the learned functionality. During this learning phase, the properties of the material evolve in response to local strains and stress on the material. This material property evolution powered by local effects on the material is the means by which a material adapts new functionalities which it hitherto did not possess. Thus, one of the hallmarks of physical learning is that it must be local. Moreover, this evolution of the properties of the material activated by local effects are often described by simple dynamic equations. Thus, the simulation of the evolution of the material properties during learning is much simpler to model and study than the optimization problems encountered in the first-principle engineering approach of material design.

A mechanical system that clearly illustrates the differences between the material design approach and the physical learning approach of material engineering is that of self-folding origamis. Origami are a simple mechanical system created from a network of stiff-less creases and stiff plastic plates circumscribed by these stiff-less creases. These initially flat 2D structures can be folded into 3D structures. The folding pathways of the flat 2D origami crease pattern into 3D object relies on the geometry of these crease patterns. It has recently been shown that certain special crease patterns/geometries result in a one-degree Maxwell of freedom system. Such a one degree of freedom origami is self-folding, that is, only the folding of one crease of the self-folding origami, results in the folding of the other creases in the origami. These 2D self-folding origamis can fold into different 3D objects, depending on the set of forces used to fold the self-folding origamis.

Another interesting property of these self-folding origamis is the glassy nature of their energy landscape. This glassy energy land scape explains why these self-folding origamis are able to fold into different 3D structures. Each of these 3D structures which these self-folding

origamis fold into represent low-energy configurations of the mechanical system. The glassy nature of these energy landscape of the origami results in so many folded modes originating from the flat mode of this self-folding origami. The number of these low-energy modes grows exponentially with the number of internal vertices/junctures of the crease pattern. These low-energy modes are also very distinct and discrete from each other. They do not flow into each other. There is no way to flow from one mode to another mode without first unfolding the origami from one folded state to the flat state, before consequently folding to a new folded state. However, the presence of these many low energy modes poses a material functionality challenge from an engineering point of view. It becomes difficult to fold an origami sheet into a desired mode without accidentally falling into an unwanted folding pathway that results in an unintended folded shape. Take for example solar powered satellites. During the travel from earth to space, the solar panels need to be folded; they unfold when the satellite arrives its final orbit destination. This is a folding problem that could be solved by engineering the solar panels into an origami crease pattern with hinges as creases. However, as previously stated, this raises an engineering challenge, as the solar panel can easily unfold into an undesired structure due to the numerous low-energy modes of the origami. Therefore, one would want to engineer a specialized origami sheet such that it only retains a single folding pathway and other unwanted folding pathways are eliminated. Such a solar panel origami pattern would be ideal for the space mission operation, as it eliminates the possibility of the solar panel unfolding into a wrong shape on arriving its destination orbit.

This engineering task to manufacture an origami pattern that only folds into a single 3D structure presents an ideal scenario to illustrate the differences between the material design approach and the physical learning approach of material engineering. It is intuitive to expect that one way to accomplish this is to bias the folding pathway of the origami by incorporating stiffnesses in the creases of the origami. It has been shown that the incorporation of stiffnesses in the origami creases can tune the energy landscape resulting in a saddle point bifurcation

which eliminates a lot of the surrounding local minima resulting in a near global minima energy landscape, which preserves the desired folding mode. Thus, from a materials design approach, the challenge lies in computing the amount of stiffness to be incorporated in each of the creases. The computation of these crease stiffnesses turns out to be a non-polynomial hard problem, which is computationally intensive. An alternative approach, the physical learning approach to material engineering, will be the subject of this work. In this work, the engineering via physical learning of a pristine origami crease pattern (which initially folded into several 3D folded structures) into a specialized origami which folds into a single 3D structure irrespective of the set of forces applied to fold it, is studied both computationally and experimentally. The work will be divided into three chapters discussing both computational simulation results and experimental results of sculpting folding pathways via physical learning in self folding origamis.

Chapter 2 discusses the theoretical and experimental aspects of physical learning in these triangulated self-folding origamis. The physical learning algorithm is guided by a simple dynamic equation, where the rate of change of the evolution of the stiffnesses on the creases of the origami is a function of the strain on the creases. The stiffness of the stiff plates of the origami remains unchanged. Parallels are established between machine learning concepts and physical learning in these origamis. Attractor-size, a measure of the performance of the physical learning is defined. This is synonymous to the cost function of machine learning algorithms. We observe undertrained, optimal trained and overtrained regimes in the physical learning of these origamis. Other parameters are defined to quantify the quality of the physical learning. Different physical learning rules – linear, quadratic, cubic and threshold – are studied and their performance and quality are recorded. This chapter also discusses the experimental studies of physical learning in disordered self-folding origamis. Single and two vertex origami crease patterns are constructed. Epoxy is used to provide an evolving stiffness that responds to the strains of the crease patterns of the origami during the learning

sessions. The origami crease setup filled with a specially prepared epoxy is mounted on a rotatory platform and trained by folding it into the desired mode repeatedly while rotating the platform to prevent the epoxy from pouring out of the creases of the origami. After training, the samples are tested with simple test forces to see how well they have learned.

Chapter 3 discusses the implications of physical learning of self-folding origamis. It introduces the achievement of supervised learning in self folding origamis via physical learning rules. In this chapter, the engineering challenge transcends from eliminating all but one low energy mode to making the origami fold into certain modes depending on the kind of forces that fold them. Basically, the origami is able to perform a classification exercise, a significant accomplishment for an unsophisticated mechanical system without "brain power." Once again, the training session is guided by dynamic physical learning rules where the rate of change of crease stiffness is proportional to the strain on the creases. The ability of dynamic local learning rules to accomplish supervised learning, as opposed to the non-local approach of gradient descent in machine learning algorithms, is the significant result of this work.

CHAPTER 2

SCULPTING FOLDING PATHWAYS THROUGH PHYSICAL LEARNING*

2.1 Introduction

[1] Metamaterials [63, 90] and smart materials [27] are often designed to show specific behaviors. For example, mechanical topological insulators localize response to forces [7, 101] while elastic networks with allostery communicate deformations over a long range [95, 116, 124]. More complex mechanical structures can show a specific deformation, e.g., a smile-shaped deformation, in response to a spatially textured pattern of forces [18]. Most commonly, mechanical systems are rationally designed to show such behaviors by searching parameter space on a computer [22, 42]. However, an alternate approach explored recently is that of *physical learning* [80, 81, 110, 105, 107]: during a period of training, the material is shown examples of the desired behavior, prompting autonomous changes in the material parameters that promote the desired behavior. No computers are involved in optimizing the properties of such a system [23, 123, 106].

Physical learning is a more constrained way of exploring parameter space than optimization on a computer. However, an autonomous physical learning process offers the advantage of learning from real examples of stimuli and response (as opposed to a theoretical specification of the problem) and does not rely on an accurate model of the material [122]. Physical learning might also allow for continual learning of new functionalities *in situ*, as requirements change [110]. A major open question is what kinds of local adaptive processes available in real systems constrain physical learning [80]. Recent work has shown that natural processes within an EVA foam network can train the network for an auxetic response by simply aging the material in different configurations [81]. Broader questions remain - what is the impact

1. *Adapted from a manuscript, Arinze et al, to be submitted.

of different kinds of local learning dynamics on the feasibility and quality of learning?

Here, we explore how the quality of physical learning depends on local adaptive processes (which we call 'learning rules') through theoretical analyses and an experimental demonstration. We focus on training a fundamentally non-linear feature in marginal mechanical systems, a bifurcation point [49, 55, 16, 8, 15, 109]. Mechanical bifurcations occur at degenerate configurations from which multiple non-linear zero modes are accessible. Bifurcations cannot be described by a linear approximation even for small deformations since the energy vanishes to two leading orders; they correspond to singularities of the energy function with an excess of linearized zero modes and self-stress states that disappear at next to leading order.

Bifurcation points can potentially be exploited to create multi-functional systems [105] and have been studied in the context of mechanical linkages [76, 62, 92] for robotics and topological meta-materials [15]. However, bifurcations can also make the system hard to control [56]. For example, folding outcomes at these singularities can be unpredictable and depend on the precise spatial pattern of forces used [109, 14, 28, 61]. While generic mechanical systems show bifurcations in some parts of configuration space [119, 120, 74], bifurcations are especially a problem for thin creased sheets because the flat state configuration is necessarily degenerate [109, 14]. In particular, creased thin sheets with nominally 1 degree of freedom (according to Maxwell counting), often called 'self-folding origami', always have a bifurcation at the flat state that is the meeting point of an exponentially large number of distinct folding modes. Consequently, such 'self-folding' sheets are hard to control at the flat state (despite the name) since the precise spatial pattern of forces applied will determine the folding geometry.

We focus on the training of such creased thin sheets [41, 111, 24] to manipulate the bifurcation point in configuration space (Fig.2.1A); in particular, we seek to increase the fraction of all force patterns that result in one folding pathway (i.e., attractor size, Fig.2.1B)

through a physical learning rule - crease softening due to folding (Fig.2.1C, Fig.2.1D). We find that successful training relies on creating heterogeneity in stiffnesses across the sheet. However, this heterogeneity can rapidly diminish with further training for some classes of learning rules while other classes of learning rules, threshold-like in strain, are more robust. We test some of these ideas with an experimental prototype in which a sheet with epoxy-filled creases is folded back and forth along one pathway at a bifurcation. Such folding during the 'training period' (i.e., before the epoxy sets) extrudes epoxy to different extents in different creases, creating a heterogeneous system. We test the trained sheet by applying different forces and find successful training in systems with 4 and 7 creases.

2.2 Results

2.2.1 Training and Stiffness heterogeneity

We begin with a theoretical study of crease patterns made of 4-valent vertices as shown in Fig.2.1A. Maxwell constraint counting gives this system 1 degree of freedom but in reality, this structure has two non-linear 1 degree of freedom pathways that meet at a bifurcation at the flat state [118]. As a toy model of energy near a generic bifurcation, consider $E(x, y) = \lambda x^2 y^2$ where x, y parameterize deformations of a mechanical system. Motions along $x = 0$ and $y = 0$ are true zero energy pathways, staying at zero energy for large deformations. However, a linearized analysis at $(x, y) = (0, 0)$ suggests a 2-dimensional vector space of zero modes (along with a self-stress state if we had a mechanistic model); further, any small deformation along $x = 0$ or $y = 0$ will reduce the zero mode space to 1 dimension (and eliminate the self-stress mode). Thus, any linearized analysis will fail to identify the true zero modes $x = 0$ and $y = 0$. On the other hand, any spatial pattern of forces applied to a 4-vertex [109] at the flat state bifurcation will result in folding along one of the two zero energy folding pathways.

The non-linear force-response relationship of disordered mechanical systems at such bifurcations can be summarized by an 'attractor diagram', shown schematically in Fig.2.1B. The space is a 2-d schematic representation of high-dimensional space of spatial force patterns. Each colored region represents one of the discrete folding outcomes that is realized for all force patterns represented by that region. Earlier work has shown that this attractor structure can be changed by changing the geometry [109, 14], pre-biasing preferred directions [115, 46, 125] of the sheet, or controlling the relative stiffness of different creases [108].

Typically, solving the inverse problem for attractors requires a complex computer algorithm; for example, Linear or Quadratic Programming algorithms on a computer [108] can determine crease stiffnesses that eliminate a chosen pathway in a saddle-node bifurcation. Here, we investigate whether the inverse problem can be solved by the sheet itself through a natural physical process, with no computers involved.

Throughout this paper, each crease, i, has a 'crease stiffness', κ_i which refers to the stiffness for the folding strain (or folding angle) about the flat state (i.e., unstrained/unfolded state) with rest angle maintained at zero; i.e., each crease, i has a folding energy $(1/2)\kappa_i\rho_i^2$ for folding strain ρ_i (see Fig.2.1D) and crease stiffness κ_i. Thus, this stiffness does not prefer any folding orientation ($\rho \to -\rho$, which is known as mountain-valley symmetry) and only depends on the magnitude of the folding strain/angle of the creases, ρ_i^2. In contrast, a creased sheet of paper develops non-zero rest angles at each crease that breaks the mountain-valley symmetry ($\rho \to -\rho$) at creases and maintains the sheet away from the flat state bifurcation, without actually reshaping the bifurcation itself [108]. In this work, we focus on the non-trivial problem of shaping bifurcations, while maintaining mountain-valley symmetry at each crease, so the trained sheet can still be completely flattened. Consequently, our trained sheet can symmetrically access either the positive or negative components of a given folding pathway.

As a first pass, we considered learning rules of the type shown in Fig.2.1C and Fig.2.1D

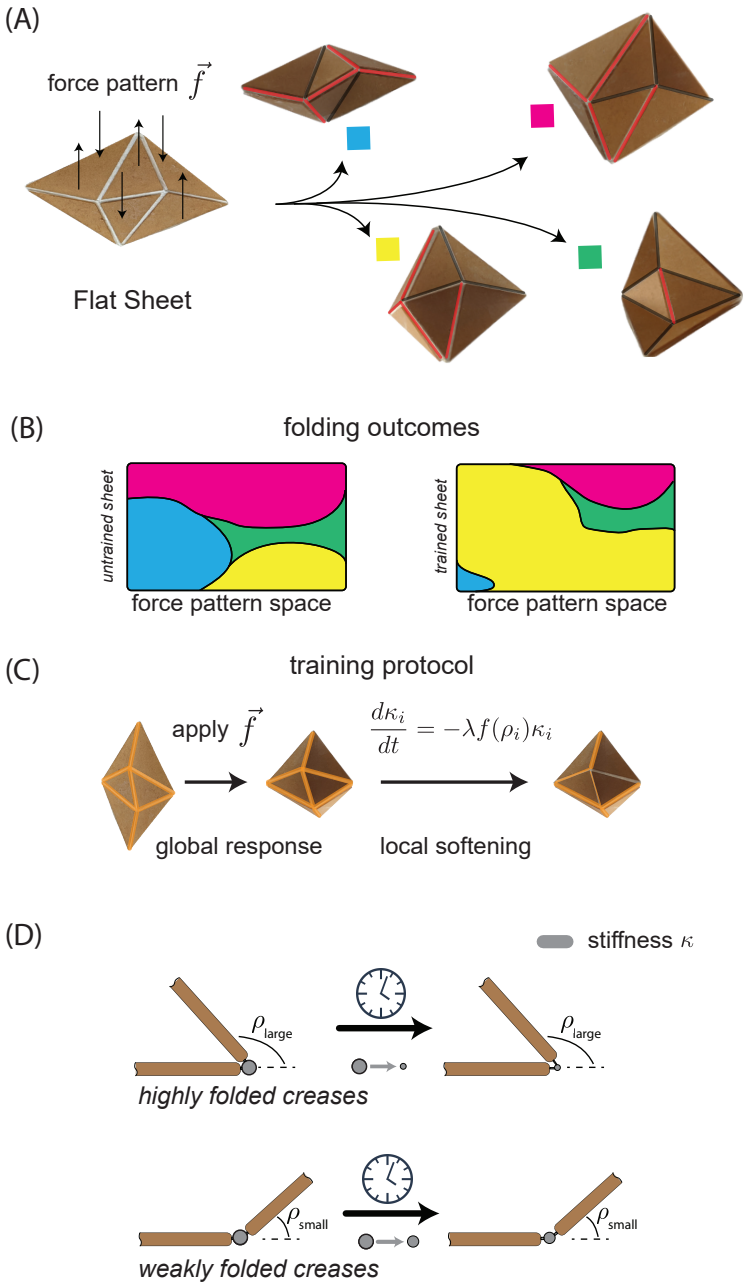

Figure 2.1: The topology of folding pathways at a bifurcation can be changed through a physical training protocol. (A) Thin creased sheets can fold along distinct folding pathways that bifurcate from the flat state with outcomes determined by the spatial force pattern. (B) A 2-d schematic representation of high-dimensional attractors in the space of force patterns. Each colored region represents one of the discrete folding outcomes that is realized for all force patterns represented by that region. Specific attractors can be enlarged, shrunk or eliminated by changing crease stiffnesses through design or physical learning. (C,D) We model a physical learning process that changes a crease i's stiffness κ_i (about the flat state) based on its folding strain ρ_i.

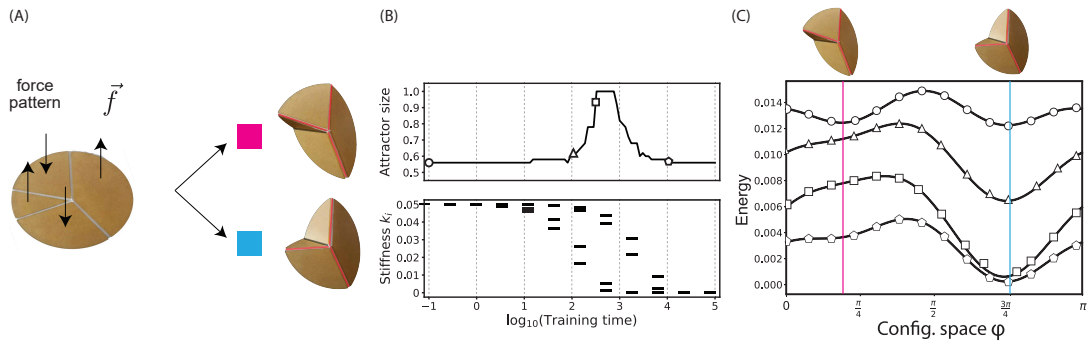

Figure 2.2: Eliminating a select folding pathway for a 4-vertex through physical training. (A) A single 4-vertex has two distinct folding modes (blue, pink). (B) The blue pathway's attractor size during a training process in which the 4-vertex is folded by a fixed force pattern repeatedly. The stiffness κ_i of each crease, i, are represented by four horizontal lines at each time point during training; all κ_i start at the same stiffness and decrease according to Eq.2.1. κ_i are in units of the bulk modulus of the stiff faces. Crease stiffnesses, κ_i, become heterogeneous during training, before becoming homogeneous again upon over-training. (C) Energy landscape of a sheet at different points during training (ϕ is an angular coordinate in 2-dimensional null space; see SI for details). Landscape before training (circles) and after over-training (pentagons) show two distinct minima; but when attractor size ~ 1 (squares), pink minimum is eliminated via a saddle-node bifurcation. Note in the under-trained regime (triangle), the attractor size is below 1, but well above the initial value of 0.56

that softens different creases based on the current folding strain:

$$d\kappa_i/dt = -\lambda\rho_i^2\kappa_i. \tag{2.1}$$

where λ, the 'learning rate', sets the learning timescale.

As a case study, we begin by applying the above learning rule to 4-vertex shown in Fig.2.2A; the 4-vertex generically has two distinct folding branches that meet at the flat state bifurcation[118]. We simulated a training process in which the vertex initially has creases of equal stiffness κ_i^0; the sheet is folded repeatedly (using a spatial pattern of forces) along the positive and negative components of one of the two pathways at the bifurcation. The sheet is folded to a configuration of finite strain and the parameters κ_i are updated according to learning rule above for a time interval τ. The sheet is then relaxed back to the flat state and re-folded with the negative of the same pattern of forces and the learning

process is continued. See SI for parameters. We test the attractor size of each pathway throughout training; the attractor size is determined by applying a library of 50 randomly chosen spatial pattern of folding torques to the creases and counting the fraction of folding torques that result in the chosen folding pathway. We assume that testing does not cause further changes in the stiffnesses κ_i in our simulations (though a real system would continue changing due to testing).

We find that during training, different creases fold to different extents ρ_i in specific ratios characteristic of the branch we fold along. Consequently, the learning rule Eq.2.1 creates heterogeneity in the initially homogeneous creases stiffness κ_i. When crease stiffnesses are relatively heterogeneous, the attractor size of the chosen pathway increases from an initial value of 0.56 to nearly 1, i.e., nearly all spatial patterns of forces result in the chosen folding pathway; see Fig.2.2B. However, further training reduces the heterogeneity in creases as all stiffnesses approach zero. In this 'overtrained' regime, the attractor size of the desired pathway drops down to 0.56 again, no bigger than for the initial untrained sheet; see Fig.2.2B.

To illustrate the phenomenon of saddle-node bifurcation, we computed the energy of the origami sheet in different folded configurations, while its crease stiffnesses,κ_i were evolved by the learning rule, Eq.2.1. We studied folded configurations within the null space of the origami's potential at the flat state. This null space is a 2-dimensional space [108]. We parameterize this null space with the variables r and ϕ. We selected a circle (defined as $r = 0.5$) in this null space and computed the energies of various folded configurations represented by points on the circumference of this circle. An energy minimum in this energy plot (Fig.2.2C) corresponds to the existence of a stable folded configuration. For untrained sheets, we see two minima corresponding to the two true non-linear folding modes. During training, the attractor size of one of these true non-linear folding modes is reduced to zero, as the mode is destroyed in a saddle-node bifurcation. This is illustrated in Fig.2.2C (see pink line), as the local minimum on the left gradually transforms to a local maximum in the

course of training. In the overtrained regime, the eliminated mode (local minimum on the left) reappears and we find that both pathways are accessible in response to some spatial patterns of forces (See images of the two modes atop Fig.2.2C).

Hence, we find that training can solve an inverse problem for non-linear behavior, namely that of eliminating one select branch in a saddle-node bifurcation and, thus, changing the topology of pathways. Further, successful training is correlated with the development of stiffness heterogeneity (see Fig.2.2B); this observation is an example of a larger principle that disordered systems can learn because the information must be stored in the trained degrees of freedom; homogeneous creases cannot store such information. Finally, we find that this particular training rule is prone to over-training and homogenization of creases if training is carried on for too long (see Fig.2.2B).

We have found that heterogeneity in the crease stiffnesses stores the learned information about the desired pathway; but the learning rule that created the desired heterogeneity also erases that information upon further training. Similar erasure of trained response was observed in systems like cyclically sheared Brownian suspensions and charge-density wave conductors [47].

2.2.2 Robustness of different learning rules

Qualitatively, many real materials soften with strain as captured by the learning rule Eq.2.1, but might differ quantitatively. Different materials will have different learning rules; some might soften proportionally to their strain or to higher powers of their strain and yet others might be even more sensitive, softening only for strain above a specific threshold. Other non-equilibrium systems can show more complex learning, where synaptic weights or learning degrees of freedom can both strengthen or soften over time; we do not investigate those cases.

We investigated whether such quantitative differences in material properties might have

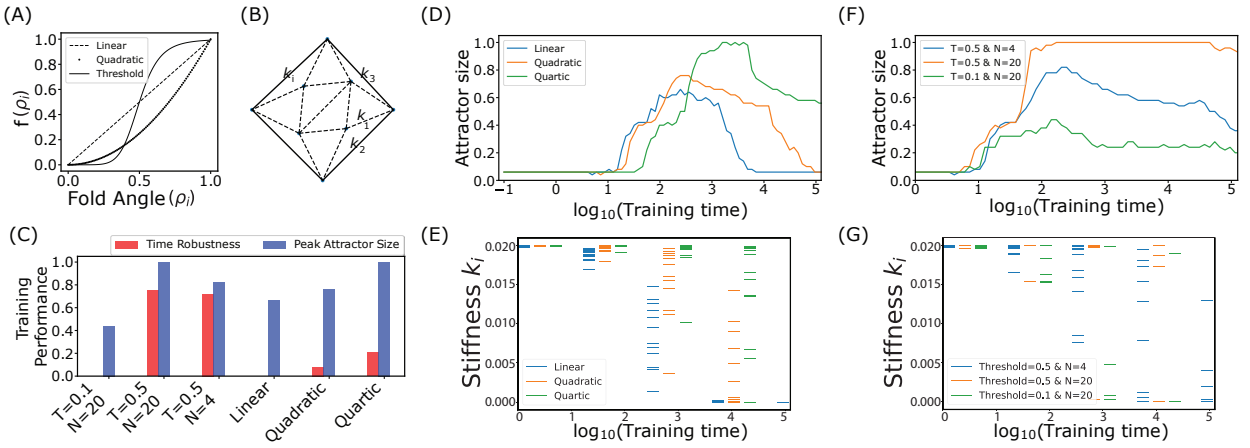

Figure 2.3: Quality and robustness of learning depend on the learning rule. (A,B) For the sheet with 13 creases shown, we model a family of 'training rules' with crease stiffness κ_i that change with strain ρ_i (which has been normalized and, thus,ranges from 0 to 1) in a polynomial $f(\rho_i) \sim \rho_i^N$ or threshold $f(\rho_i) \sim \rho_i^M/(\rho_i^M + T^M)$ manner. (C) Learning quality (peak attractor size; see D,F) and robustness (length of training time over which attractor size > 0.80; see D,F) for different learning rules are explored here. (D) With $f(\rho_i) \sim \rho_i^N$ rules, the attractor size of the desired pathway rises to a peak before falling. The peak attractor size (quality of learning) increases with N. (E) The stiffness of each of the 13 creases, i, of the origami are represented by the 13 horizontal lines. They all start out with the same stiffness and decrease according to the different polynomial learning rules indicated by the colors of the line. κ_i are in units of the bulk modulus of the stiff faces. The polynomial rules also show a regime of over-training where stiffness κ_i become homogeneous again as all κ_i go to zero. The horizontal lines representing the crease stiffnesses for the different polynomial learning rules are plotted at select times, so as to avoid the lines of the different polynomial learning rules from overlapping. (F) Attractor size vs training time for threshold learning rules. (G) The stiffness of each of the 13 creases, i, of the origami are represented by the 13 horizontal lines. They all start out with the same stiffness and decrease according to the different threshold learning rules indicated by the colors of the line. κ_i are in units of the bulk modulus of the stiff faces. Crease heterogeneity grows with training, and does not decrease (no overtraining) with continued training for some of the threshold learning rules (orange line).The horizontal lines representing the crease stiffnesses for the different threshold learning rules are plotted at select times, so as to avoid the lines of the different threshold learning rules from overlapping.

a qualitative effect on the quality and robustness of learning. We considered families of rules of the type:

$$d\kappa_i/dt = -\lambda f(\rho_i)\kappa_i. \tag{2.2}$$

The first family we considered were different polynomial forms $f(\rho) \sim \rho^N$. Note that Eq.2.1 is the case where N=2 (See Fig.2.3A). The second family, defined by a Hill coefficient M, $f(\rho) \sim \rho^M/(\rho^M + T^M)$ corresponds to sigmoidal dependence often seen in real systems; small strains do not cause significant aging or change in stiffness but strains above a characteristic threshold T cause stiffness changes; further the precise amount of strain does not matter beyond this threshold T (See Fig.2.3A).

We trained a larger disordered creased sheet with 13 creases and 4 vertices (See Fig.2.3B); this sheet had 16 non-linear modes meeting at the flat state bifurcation. We used the same training protocol as for the single vertex in Fig.2.2: we selected one pathway as the desired pathway and applied the learning rule as the structure was folded repeatedly into the positive and negative components of the selected pathway. As earlier, we quantified successful learning by the attractor size of the desired pathway, i.e., folding the sheet with a library of 50 random force patterns and computing the fraction of force patterns that result in a specific folded mode. We assume that stiffnesses do not change during such testing.

Among polynomial learning rules $f(\rho) \sim \rho^N$, we found that training rules with higher order polynomials (higher N) resulted in better training. We found a higher peak attractor size and training was also more robust - learning quality stayed higher for longer (See Fig.2.3C). However, all polynomial learning rules were still susceptible to overtraining, during which the crease stiffness heterogeneity was lost (See Fig.2.3D, Fig.2.3E). With threshold-like learning rules, the over-training problem was nearly eliminated (See Fig.2.3F). We found that creases that fold less than the threshold T do not soften at all and hence the learned heterogeneity in stiffness is maintained over time. However, there is a trade-off; the threshold T of the learning rule needs to be within the range of strains experienced during training. If T is

too large, no training would occur. If T is too small, training will fail to create sufficient heterogeneity in stiffnesses κ_i to encode information about the desired mode (See Fig.2.3G).

2.2.3 Experimental demonstration

With these theoretical results in place, we demonstrated these ideas with an experimental prototype. While many previous works have implemented creased sheets in systems ranging from graphene on the nanoscale [70] to mylar and cardboard on the mesoscale [87] to solar panels on satellites [71], these works generally have fixed stiffness in different creases and thus an inability to learn folding behaviors. We note that paper or cardboard is affected by folding but typically develops a non-zero rest angle upon folding and is thus likely to fold in the same way again. But as noted earlier, such a displacement from the bifurcation does not reshape the bifurcation which still exists if the sheet is forced into the flat state. Here, we create a prototype that maintains $\rho \to -\rho$ symmetry at each crease and hence can still be laid flat after training.

We created a sheet with gullies at creases by sandwiching a Tyvek sheet between acrylic pieces that were laser cut to serve as stiff faces of a crease pattern. See Fig.2.4A. Consequently, the creases correspond to gullies of width w (set by the gap between adjacent acrylic pieces) and depth h (set by the thickness of acrylic) on both sides of the sheet. A slow setting epoxy solution is created from a mixture of epoxy resin and a curing agent in the ratio $1:2$. The creases are filled with epoxy on both sides of the sheet; the epoxy takes ~ 60 minutes to set. See SI for details. During this setting time (the 'training phase'), the crease is folded to an angle $+\rho$, flattened again and folded to angle $-\rho$ in the other direction. Such folding will extrude an amount of epoxy $h(\rho)$ from the crease gully that depends on the magnitude of folding $|\rho|$. Only epoxy remaining inside the crease determines the stiffness of that crease at the end of the training phase. Thus, the amount of epoxy extruded $h(\rho)$ determines the change in stiffness $\Delta\kappa$ during training and thus determines the form of the learning rule in

Eq.2.2. By folding in both directions ($\pm\rho$) during training, we maintain mountain-valley symmetry and zero rest angles at the crease; consequently, the trained sheet can still be laid flat.The training protocol is illustrated in Fig.2.4B.

We constructed a vertex with 4 creases (studied theoretically in Fig.2.1), resulting in two folding modes (pink and blue) shown in Fig.2.4C. Initially, before any epoxy is added, all creases are free folding. We tested the response of this free folding vertex to forces applied at different points along the boundary of one of the sectors. We found that forces at 3 of 7 locations leads to the blue folding mode while forces at the 4 locations result in the pink mode (See Fig.2.4C). Forces applied to all points in the other sectors led to either the blue or the pink mode. We then filled the creases with slow setting epoxy, marking the start of the training phase. We folded the vertex into a selected mode (blue) with characteristic folding angles ρ_i^{blue} at each crease, reverted to the flat state and folded along the negative component $-\rho_i^{blue}$ of the same blue pathway. We folded until the largest folded crease could not be folded further; in this way, the magnitude of folding is approximately the same along the positive and negative components of the pathway and over multiple instances of folding during training. Throughout this training, the vertex was clamped in a vertical configuration but was rotated periodically to prevent epoxy from flowing out of the creases due to gravity. We repeated this folding process for 60 minutes, folding back and forth along the positive and negative components of the blue mode. See SI for details of the experiment.

After the epoxy set, we found that forces at all 7 test locations now lead to the blue mode as illustrated in Fig.2.4C, showing that the training procedure had modified the flat state bifurcation, presumably by eliminating the pink branch at a saddle-node bifurcation away from the flat state [108]. Using a fresh sample, we also successfully repeated the training process above to retain the pink mode and eliminate the blue mode instead.

To see if the principles behind this simple demonstration are robust enough to work in more complex disordered systems, we attempted training on a sheet with 7 creases, 2

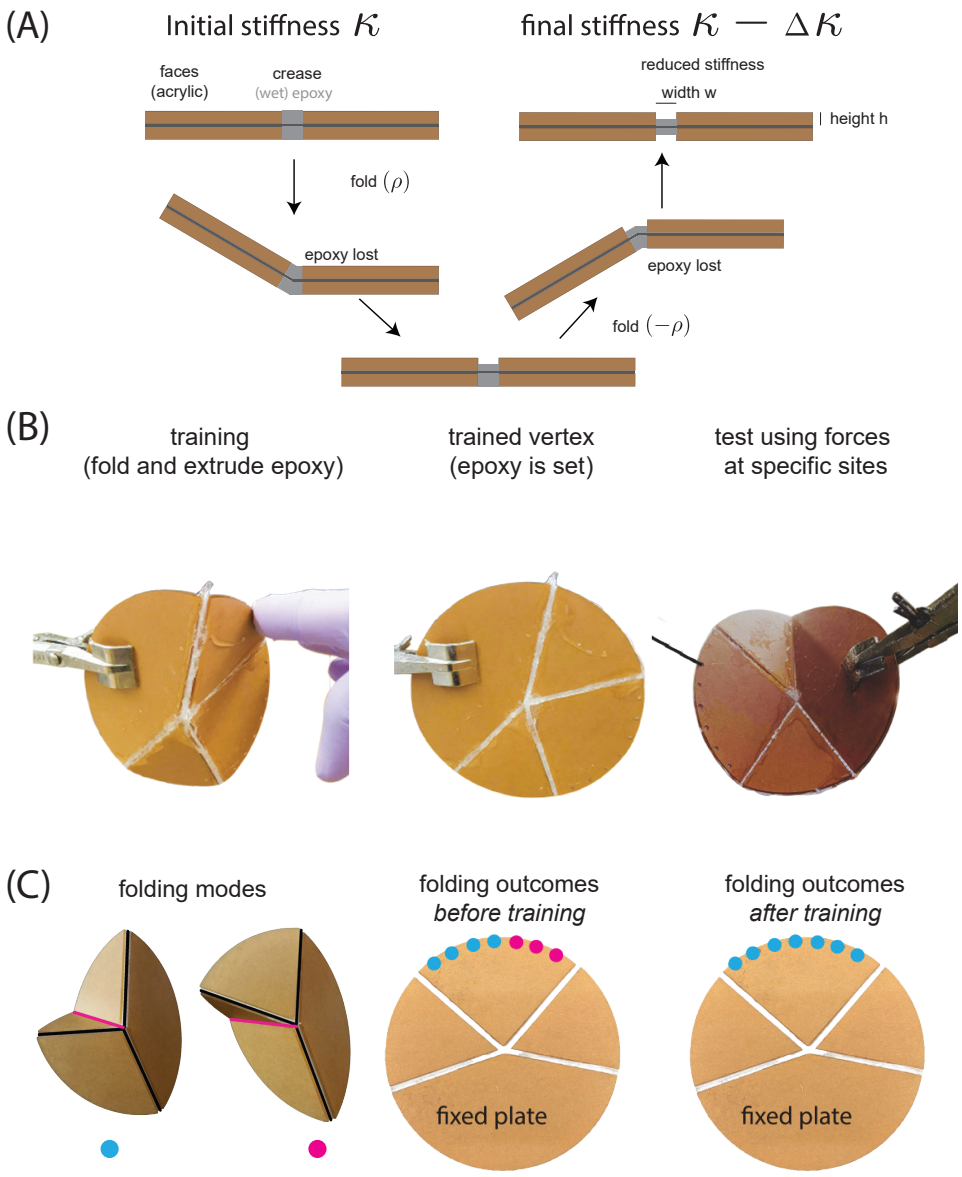

Figure 2.4: Experimental realization of training through epoxy extrusion. (A) (schematic) We created creased sheets by sandwiching a thin membrane between thick acrylic sheets that serve as stiff faces; the resulting creases of width w and height h are filled with epoxy (orange). Folding by angle ρ before the epoxy sets (the training period) will extrude epoxy; creases with larger ρ will extrude more epoxy, resulting in lower stiffness κ after the epoxy sets. (B) A 4-crease vertex with epoxy filled creases was trained by repeatedly folding along one of the two pathways during the setting period; folding is repeated along positive and negative components of the chosen pathway to avoid any directional folding bias in the creases. (C) Testing: Folding outcomes are determined for folding forces applied at different locations. The untrained sheet folds along two distinct pathways depending on location of applied force (pink and blue dots). The trained sheet folds along only one pathway for all folding forces (blue dots), demonstrating an increased attractor for the blue pathway.

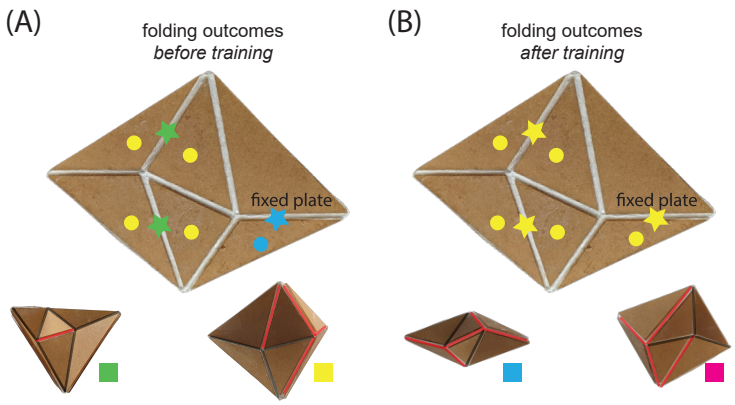

Figure 2.5: Learning to expand a select attractor in a complex sheet. A sheet with 7 creases and 2 vertices has 4 distinct folding pathways (shown at the bottom). (A) Before training, three pathways are accessible by forces applied to different locations shown (circles) or by torques applied to specific creases (stars). (B) The creases were filled with epoxy and folded back and forth along the yellow pathway as the epoxy set (the training period). After training, the sheet was 'tested' with the same forces and torques used in (A). All test forces and torques now result in the yellow pathway, indicating an expanded attractor size for that pathway.

vertices and thus 4 distinct pathways at the bifurcation; see Fig.2.1A. As shown in Fig.2.5A, the untrained sheet folds along 3 of those 4 pathways for test forces applied to the center of different faces with one face held clamped. (The fourth pathway requires torques at specific creases that cannot be realized by forces at a single face in the clamped configuration we used.) We filled the creases with epoxy and trained with the same protocol as earlier, folding along a select pathway (the yellow pathway), flattening the sheet, folding along the negative branch of that pathway and repeating the process for 60 minutes. After the training process is completed and the epoxy has set, we tested the sheet with the same test forces applied to the faces when it was untrained; we now find that all forces lead to folding along the target yellow pathway as illustrated in Fig.2.5B. Thus, the flat state bifurcation has been successfully trained to eliminate the other pathways, presumably through saddle-node bifurcations away from the flat state.

2.3 Discussion

The study of bifurcations in mechanical systems has attracted attention from mathematicians [45], roboticists [119, 120] and physicists [14, 109, 108]. Most work has focused on changing the structure of bifurcations by rationally changing parameters such as lengths of the elements. Our work here shows at least some versions of this design problem can be solved by changing stiffness of joints through a physical training protocol. Our work further suggests that bifurcations might be physically trainable in mechanical linkages where the lengths of elements change according to learning rules; changes in length have been used as a basis for physical training in other contexts [81, 38, 37, 107].

The experimental demonstration here illustrates how a generic physical process - the extrusion of material at creases - can naturally implement 'learning rules' that confer specific functionality on the system. Other materials naturally show softening with strain [81], possibly allowing for the implementation of different functional forms of our learning rules.

The locality of physical learning in mechanical systems contrasts the global nature of most machine learning algorithm, where learning parameters are non-locally updated (gradient descent protocols). The training rules are local, softening each crease in response to strain in that crease. Further, even at the end of training, the learned stiffness in any one crease does not immediately favor one folding pathway over another. However, creases that meet at a vertex [87] have non-linear interactions that constrain their relative folding; such interacting creases are able to collectively learn and encode information about a desired pathway even if each individual crease does not select a pathway by itself.

While we trained for one attractor to grow and occupy most of the force pattern space, one can also train a system for multiple attractors [105]. Such training can create a mechanical pattern recognizer, folding into one configuration in response to one set of force patterns and a distinct configuration in response to a different set of force patterns. Unlike similar responsive materials designed on a computer, the learning paradigm here lets structures learn

in situ from real examples of force patterns [81].

2.4 Supplementary Note

2.4.1 Supplementary Note 1 - Theory

Theoretical Modeling of Self-Folding Sheets

We model creased sheets using energy-based models developed in previous work [41, 112, 87]. We assume that creases have a stiffness modeled by torsional spring elements on each crease [109, 108, 105]. We briefly review the elements of this model as we build upon this work to simulate the physical learning of desired folding pathways of self folding origami sheets.

The energy of thin sheet origami is dominated by face bending governed by mechanical constraints at the origami vertices. Each vertex contributes 3 constraints on the folding angles of creases around it [112]. Take a vertex surrounded by N creases denoted with an index i and each folded to an angle ρ_i. At the flat state, all $\rho_i = 0$, which trivially satisfies all mechanical constraints. One can write down an expansion for these 3 non-linear constraints $T_a(\rho_i)$ [109]. The energy of the folded origami is taken as the sum of squares of the residues of these constraints $E_{\text{Vertex}} \sim \sum_a T_a^2$, which is independently summed over different vertices [108]. The energy due to the stiffness in the creases is $E_{\text{Crease},i} = \frac{1}{2}\kappa_i \rho_i^2$ as discussed in the main text. Thus, the total energy of a folded sheet is the sum of vertex and stiff crease energies

$$E_{\text{sheet}}(\rho_i) = \sum_{v \in vertices} \sum_{a=1}^{3} T_{va}^2 + \frac{1}{2} \sum_{i \in \text{creases}} \kappa_i \rho_i^2. \tag{2.3}$$

In the learning protocol presented in this work, the crease geometry is fixed and so are the vertex constraints T_{va}. The change in the energy of a folded configuration ρ_i during

training stems directly from the change in the individual crease stiffness values

$$\frac{dE}{dt} = \frac{\partial E}{\partial \kappa_i}\frac{d\kappa_i}{dt} = \frac{1}{2}\rho_i^2\frac{d\kappa_i}{dt}. \tag{2.4}$$

Simulated Sheet Folding

Using the energy model described previously, we simulated the folding of the self folding origami via several numerical folding methods [108].

Origami sheets are numerically folded similarly to the way described in [105]. The folded sheet's configurations are initialized at

$$\rho_{i0} = \rho\frac{\tau_i}{||\tau_i||}$$

by a set of external folding torques (τ_i) on the creases with $\rho \equiv ||\rho_i||$ the folding magnitude, chosen to be $\rho = 0.5$. However, this initialization point is typically not an energy minimum on the surface of the hyper-sphere of radius ρ. Thus, we numerically relax the sheet to a local minimum of Eq.3.7 using MATLAB, subject to a constraint that fixes the folding magnitude ρ:

$$\begin{aligned} \underset{\rho_i}{\text{minimize}} \quad & E_{\text{sheet}}(\rho_i) \\ \text{subject to} \quad & ||\rho_i|| = \rho. \end{aligned} \tag{2.5}$$

This protocol mimics the experimental fast folding of origami sheets, and the clamping of one crease at a specific folded dihedral angle. It was tested and validated in [108]. The results of this folding protocol are similar to torque based folding of sheets using Newtonian methods.

Simulation of Sculpting Folding Pathways Through Physical Learning

Starting with Eq.2.4 above we see that the evolution of the energy landscape and, thus, folding pathway is driven by $\frac{d\kappa_i}{dt}$. Physical learning is introduced by the dynamic specification of $\frac{d\kappa_i}{dt}$ defined by various physical learning rules, which are functions of the fold angles of the mode $\rho_i^{Teacher}$, whose attractor size we desire to expand:

$$d\kappa_i/dt = -\lambda f(|\rho_i^{Teacher}|)\kappa_i. \tag{2.6}$$

where λ, the 'learning rate', sets the learning timescale and $\rho_i^{Teacher}$ is a vector defining the fold angles of the creases of the desired mode whose attractor size we want to increase. Note that $\rho_i^{Teacher}$ is obtained by folding the creases of the origami with an external torque, $F_i^{Teacher}$ as described above. We generated the components of $F_i^{Teacher}$ by first randomly selecting a number from a normal distribution. Next we normalized this vector and used it to fold the creases of the origami as described above. We checked if the scalar product between the normalized vector and the resulting normalized folded mode is greater than 0.99. If it is not, we generate another $F_i^{Teacher}$ by selecting another random set of numbers for its components and check if the new vector and its resulting folded mode have a scalar product greater than 0.99. If the scalar product is greater than 0.99, then the final resulting folded mode is then normalized and assigned to $F_i^{Teacher}$. The learning rule is specified by $f(\rho_i^{Teacher})$ which can be a linear, quadratic or a threshold function of $\rho_i^{Teacher}$. We simulated a physical learning process in which the creases initially had a uniform stiffness $\kappa_i^0 = 0.02$ (a unit of stiffness represents the bending modulus of the stiff faces of the origami) and evolved with a learning rate $\lambda = 0.01$ per training round. Thus, we have specified not just a teacher for the physical learning process, $F_i^{Teacher}$, but also a curriculum or learning rule, $f(\rho_i^{Teacher})$, for the physical learning process.

Simulation of Testing Protocol

After each round of training via physical learning as described above, the attractor size of the desired mode is computed. To calculate the attractor size, a set of an array of test torques, $TestTorqueSet$ is created. Each element of the set, $TestTorqueSet$ is defined by a vector $F_i^{external}$, whose components, $F_{i,j}^{external}$, represent the magnitude of the torque applied to each crease. Each vector $F_i^{external}$ is normalized and used to fold the creases of the self folding origami sheet as described above. Folding with each external torque, $F_i^{External}$, results in a folded mode ρ_i^{Folded}. Note that the folded mode vector, ρ_i^{Folded} is also normalized as well. This folded mode, ρ_i^{Folded} is compared to the desired mode, $\rho_i^{Teacher}$ whose attractor size we seek to increase.

To compare the folded mode, ρ_i^{Folded}, with the desired mode, $\rho_i^{Teacher}$, we take the scalar product between both vectors. If the scalar product between the two vectors is above 0.9, then we consider the two modes as similar, and one and the same. We count the number of external torques, $F_i^{External}$, in the set $TestTorqueSet$, whose folded mode, ρ_i^{Folded}, are considered similar to $\rho_i^{Teacher}$. We then express this count number as a fraction of the cardinality of the set $TestTorqueSet$. This fraction defines the attractor size of the desired mode.

Quality of Learning

Two agents drive the physical learning process: the teacher, $F_i^{Teacher}$, and the curriculum/learning rule, $f(\rho_i^{Teacher})$. We found the kind of teacher selected does not affect the quality of the physical learning as long as it results in the desired folded mode. Hence, the quality of learning is determined by the kind of learning rule selected. We quantify the quality of learning for various learning rules with two parameters: the peak attractor size attainable and the time robustness of the learning rule. The peak attractor size compares the maximum attractor size achieved for a desired mode for the different learning rules.

Meanwhile, the time robustness measures the percentage of the training round for which the self folding origami is optimally trained (i.e. attractor size is above 0.80). The time robustness is a measure of the training protocol's resilience against over-training.

Calculation of Energy Landscape for A Single Vertex Origami

To illustrate the mechanism by which physical learning alters the energy landscape of the self folding origami via a saddle-node bifurcation, we plotted the energy, Eq.3.7 of the different folded configurations at several points during the training of the origami. After each round of training, the stiffnesses in the creases of the single vertex changes and the energy landscape of the folded configuration space is re-computed.This time-energy landscape plot shows the elimination of the unwanted folding pathway (mode) via a saddle-node bifurcation, and the preservation of the desired folding pathway (mode) after several rounds of training. Further training results in a recovery of the previously eliminated mode.

2.4.2 Supplementary Note 2 - Experiments

Acrylic sheet setup

To create a system naturally capable of learning, we exploited an origami system with fresh epoxy totally filled into the crease pattern of the origami. This epoxy is extruded from the creases during the folding of the origami during the training protocol. This results in a final stiffness (after the epoxy sets) that depends on the amount of folding of each of the creases. We laser cut origami patterns in acrylic sheets of thickness 1.5mm; crease lines were designed to have a gap (or width) of 30mm. Test holes of 10mm diameter are laser cut on the acrylic sheet at various strategic positions (along the circumference of the 90° plate for the single vertex and on the center of each plate of the two-vertex). Two copies of such acrylic patterns were each glued to both sides of a sheet of Tyvek.The corresponding plates of the origami

patterns on each side of the sheet of Tyvek are lined up with each other before the glue sets. After the glue is set, holes for applying testing forces on the faces of the origami plates are perforated. The resulting setup has stiff faces (bending stiffness set by acrylic) and soft creases (stiffness set by the Tyvek sheet). Origami patterns studied were for a single-vertex and two-vertex. The single vertex has four creases radiating from a single vertex at the center of the pattern. The creases of this single vertex pattern form sector-angles 150°, 60°, 90°, and 60°. The two-vertex pattern with a total of 7 creases consists of two internal vertices; one vertex has the following sector angles 107°, 123°, 82°, and 48°, while the second vertex is surrounded by sector angles 82°, 54°, 99°, and 125°. The two vertices are connected by a common crease. This connecting crease serves as the boundary dividing the 82° sector plate from the 48° sector plate of the first vertex, and the 82° sector plate from the 54° sector plate of the second vertex. The sector angles, crease lengths, and position of the vertices for both the single-vertex and two-vertex are specified in a PDF file which can be used to laser cut these patterns.

Epoxy and training

Epoxy Mixture An epoxy solution is made by mixing epoxy resin with its curing agent in the ratio 1:2. This mixture is stirred for about five minutes and poured into the creases of the origami pattern on both sides of the assembly. Note, that if the epoxy had been mixed in the ratio 1:1, upon curing, it would be so stiff that the origami assembly would be difficult to fold, without destroying the assembly; such epoxy when hardened is also brittle and would fracture under a bending moment attempting to fold the origami assembly. Hence, we mix epoxy resin and curing agent in the ratio 1:2, allowing for crease folding upon curing of the epoxy mixture, without disintegration of the origami assembly.

Folding of Origami Assembly The origami assembly filled with watery epoxy in its creases is folded into the desired configuration that is to be trained for. The origami assembly is trained by folding the assembly back and forth along the positive and negative components of a desired folding pathway. This cyclic folding between both pairs of the desired mode is repeated for an hour, during which the epoxy solution begins to cure and is no longer watery. A simple folding protocol is utilized to fold the origami assembly into the desired configuration: one of the plates of the origami assembly is fixed while pushing or pulling on any of the other plates of the origami assembly with a normal force or a turning torque exerted at a single contact point on any of the non-fixed origami plates.

Training Under Gravity Since the epoxy is still watery during origami training, it needs to be trained on a rotating platform to avoid epoxy from flowing out of the creases due to the influence of gravity. The rotating platform consists of two standing laboratory clamps screwed to the optical table and situated 600mm apart. A rod is horizontally supported by the claws of the two standing laboratory clamps, but the rod is allowed to freely rotate within the claws of the clamp. A lab clamp retort is then fixed clamped (allows for no rotation or slipping) to the rotation-free horizontally placed rod with the claws on one end of lab clamp retort, while the claws on the other end of the lab clamp retort is clamped to one of the plates of the origami (the plate fixed during training as previously described). As one hand is used to exert a normal force or turning torque on one of the free plates of the origami assembly in order to fold it into the positive and negative components of the desired mode, the other hand is used to manually turn the rotation-free horizontal rod. This combined process results in the folding of the origami assembly, while under rotation, and, thus, prevents the flow of the epoxy solution from the origami's creases during this training; while under the influence of gravity.

Emergence of Crease Stiffness

After training the origami assembly, the origami samples are allowed to further cure and are left hanging in the lab for a week, allowing the epoxy solution in the creases to harden, thus, producing an effective stiffness on the creases of the origami.

Testing protocol

The plate of the origami assembly fixed during training is clamped. A 100mm long thread knotted on one end is passed through each of the testing holes of the origami assembly. The threads are gently pulled normal to the surface of the origami plate. Upon pulling each thread in each hole, the origami folds into a given configuration. The resulting configuration for each pull is recorded. The attractor size of the different folding modes of the origami assembly is computed. This process is repeated for both the trained and untrained samples of the origami assembly. The attractor size of the chosen trained mode before and after training are compared to one another.

CHAPTER 3

SUPERVISED LEARNING THROUGH PHYSICAL CHANGES IN MECHANICAL SYSTEMS*

3.1 Introduction

[1] Mechanical metamaterials are usually designed to show desired responses to prescribed forces. In some applications, the desired force-response relationship is hard to specify exactly, but examples of forces and desired responses are easily available. Here we propose a framework for supervised learning in thin creased sheets that learn the desired force-response behavior by physically experiencing training examples and then, crucially, respond correctly (generalize) to previously unseen test forces. During training, we fold the sheet using training forces, prompting local crease stiffnesses to change in proportion to their experienced strain. We find that this learning process reshapes non-linearities inherent in folding a sheet so as to show the correct response for previously unseen test forces. We show the relationship between training error, test error and sheet size (model complexity) in learning sheets and compare them to counterparts in machine learning algorithms. Our framework shows how the rugged energy landscape of disordered mechanical materials can be sculpted to show desired force-response behaviors by a local physical learning process.

The design of mechanical metamaterials usually assumes that desired force-response properties are given as a top-down specification. For example, principles of topological protection can be used to design materials where forces at specific sites lead to localized deformations [77, 7] while other principles [95] can help communicate that deformation to a specific distant site. In these examples and many others [78, 51, 50, 86, 108], we rationally optimize design parameters, e.g., spring constants and geometry, to achieve a specified force-response relationship.

1. *Adapted from Stern, Arinze et al, PNAS 2020

A different approach, closely connected to supervised learning in computer science, is useful when the force-response behavior is too complex to specify in a top-down manner but it is easy to give examples of the desired behavior. In this learning approach, the emphasis is on inferring the right force-response relationship from such training examples, with success evaluated on the ability to extrapolate the relationship to unseen test examples.

In this work, we wish to employ the advances of learning theory but, crucially, perform the learning outside of the computer, at the level of the *physical* material itself. This way, we introduce a *physical learning model*, able to learn from (adapt to) shown examples and generalize to novel, unseen examples. The ability to *generalize* to show the correct response to novel test examples has fueled the success of machine learning. Such generalization would also be useful in materials when the use cases are either complex or are not fully known at the time of design. Consider applications such as Micro-electro-mechanical systems (MEMS), where a mechanical membrane must either complete or open an electrical circuit by deploying one of two folding responses A or B in response to two different sets of force patterns S^A or S^B respectively.

Only some examples of the force patterns in the sets S^A, S^B may be known at the time of design. Further, the distinction between the force patterns in the sets S^A, S^B may not be easily apparent, similarly to how pixel intensity correlations that distinguish images of cats and dogs are non-trivial. However, much in the way a neural network can learn correlation features that distinguish cats from dogs by seeing examples, a training procedure for materials might be able to learn and respond to high-dimensional correlation features that distinguish force patterns in the sets S^A and S^B. Crucially, the trained material could distinguish novel patterns in S^A, S^B that were not used as part of the training. Finally, even when distinguishing features are known *a priori*, learning offers a natural way for materials to arrive at the right design parameters themselves, without need for a complex optimization algorithm on a computer.

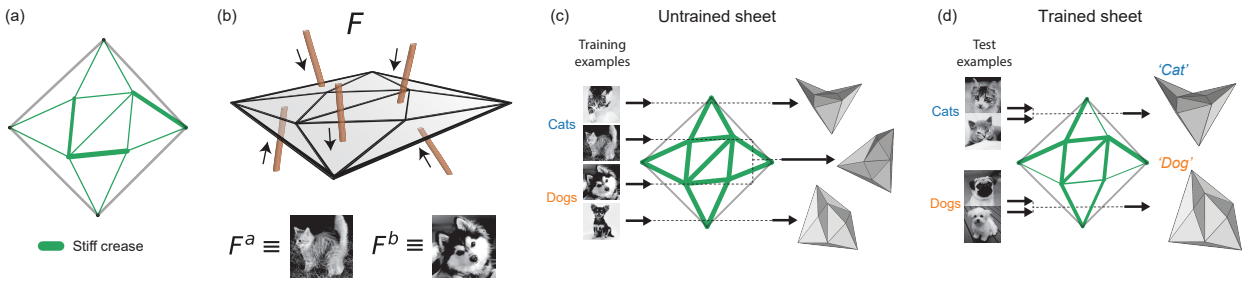

Figure 3.1: Training thin sheets to classify spatial force patterns. a) We consider thin stiff sheets with creases whose stiffnesses (indicated by thickness of green segments) can be changed by repeated folding. b) Such sheets can fold in response to a discrete spatial force pattern $\mathbf{F}$ applied on the sheet. To emphasize the high dimensional nature of $\mathbf{F}$, we draw an analogy between $\mathbf{F}$ and an image where the gray-scale value of each pixel corresponds to the force at a particular location on the sheet (see Supplementary Note 1). Particular force patterns correspond to examples of different classes (e.g. cat $\mathbf{F}^a$) and dog $\mathbf{F}^b$). c) An untrained sheet with uniform stiffness shows random folded responses for different spatial force patterns. d) By allowing crease stiffness to change in response to strain, we train the sheet to learn correlation features that distinguish different classes of force patterns. Consequently, the sheet classifies force patterns by showing one distinct folded response for each class.

While naturally occurring systems like neural networks [30], slime molds [1], and plant transport networks [99] use similar ideas to adapt their response to environmental inputs, mechanical supervised learning has thus far not been used to obtain functional man-made materials. Here we propose an approach for the supervised training of a mechanical material through repeated physical actuation. We work with a model of creased thin sheets where crease stiffnesses can change as a result of repeated folding. We assume a training set, that is, a list of force patterns and desired responses. Each training example of force pattern is applied to the sheet; if the response is the desired one, as determined by a 'supervisor', folded creases are allowed to soften in proportion to their folding strain. If the response is incorrect, creases stiffen instead. We then test the trained sheet by applying unseen force patterns (test examples) drawn from the same underlying distribution as the training data. We study test and training errors and thus the sheet's ability to generalize to novel patterns as a function of its size.

Our proposal here relies on a plastic element, namely crease stiffness. Materials that stiffen or soften with strain have been demonstrated in several contexts [73, 88, 64, 2], including recently in the training of mechanical metamaterials [81]. We discuss how learning performance may be affected by limitations on the dynamic range of stiffness and other practical constraints in such materials. We hope our results here will provoke further work on how the frameworks of learning theory could inform the creation of new classes of designer materials.

3.2 Results

We demonstrate our results with a creased thin self-folding sheet (Fig. 3.1a), which is naturally multi-stable. Our analysis can be generalized to other disordered mechanical systems, such as elastic networks [81], that are also generically multi-stable. It was previously shown that creased sheets, such as those of self-folding origami, can be folded into exponentially many discrete folded structures from the flat, unfolded state [109, 14]. Such exponential multi-stability can be a problem [109, 108] from an engineering standpoint, as precise controlled folding is required to obtain the desired folded structure.

Here we exploit such multi-stability to train a classifier of input force patterns. If we apply a spatial force pattern F on the flat sheet (see Fig. 3.1a-b), the sheet will fold into a particular folded structure $\rho(F)$, e.g., described by dihedral folding angles at each crease ρ_i. To emphasize the high dimensional nature of space of force patterns, in Fig. 3.1b-d, we represent each force with a gray-scale image where pixels values are an analogy to forces at designated points on the sheet. In practice, we apply these forces as torques directly on creases (see Supplementary Note 1). The set of all force patterns $\{F\}$ that lead to one particular folded structure ρ^m, due to the dynamics of folding, is defined as the 'attractor' of folded structure m in the space of force patterns (color coded regions in Fig. 3.2b). The complex attractor structure of force-response for a thin sheet naturally serves as a classifier

of force patterns, albeit a random classifier (Fig. 3.1c). The goal of the training protocol is obtain a sheet with a specific desired mapping between force patterns and folded structures (Fig. 3.1d).

The mapping between force and folded structure is controlled by local properties of the system, such as thickness or stiffness [17, 46] (See Supplementary Note 1). Previously, we found that the folded response to a given force pattern F can be modified by changing the stiffness k_i of different creases i in the sheet [108]. Here, we employ a 'supervised learning' approach to naturally tune stiffness values k_i so that the sheet classifies forces as desired. Intuitively, this is done by applying examples of force patterns to the sheet and modifying crease stiffness accordingly, in a way that reinforces the correct response and discourages incorrect folding (Fig. 3.2a). Such training, carried out iteratively for different force pattern examples, has the effect of morphing the attractor structure to better approximate the desired response (Fig.3.2b).

Consider two distributions of force patterns, each designated as a particular class (e.g. 'cats' and 'dogs'). For example, Fig. 3.3a (top) shows two classes of forces defined by: $S^{\text{dog}} = \{F | F \cdot F_{\text{dog}} \geq D, F \cdot F_{\text{dog}} > F \cdot F_{\text{cat}}\}$, and similarly for S^{cat} for a threshold $D = 0.6$. (In Fig. 3.3a, S^{dog} is blue and S^{cat} is orange.)

Assume we are given two sets of labeled force patterns as training examples $\mathcal{F}^{\text{dog}} = \{F \in S^{\text{dog}}\}$, $\mathcal{F}^{\text{cat}} = \{F \in S^{\text{cat}}\}$, each with n training force patterns (in Fig. 3.3a (bottom) we sample sets with $n = 20$). Together, $\mathcal{F}^{\text{dog}}$ and $\mathcal{F}^{\text{cat}}$ are defined as the *training set*. We desire all forces in S^{dog} to result in one common folded structure, while all forces in S^{cat} fold the sheet to a distinct but common folded structure. While S^{dog}, S^{cat} are separable in some $2d$ projection of force space, learning is non-trivial since the sheet must learn the 2 dimensions in which these distributions are separable.

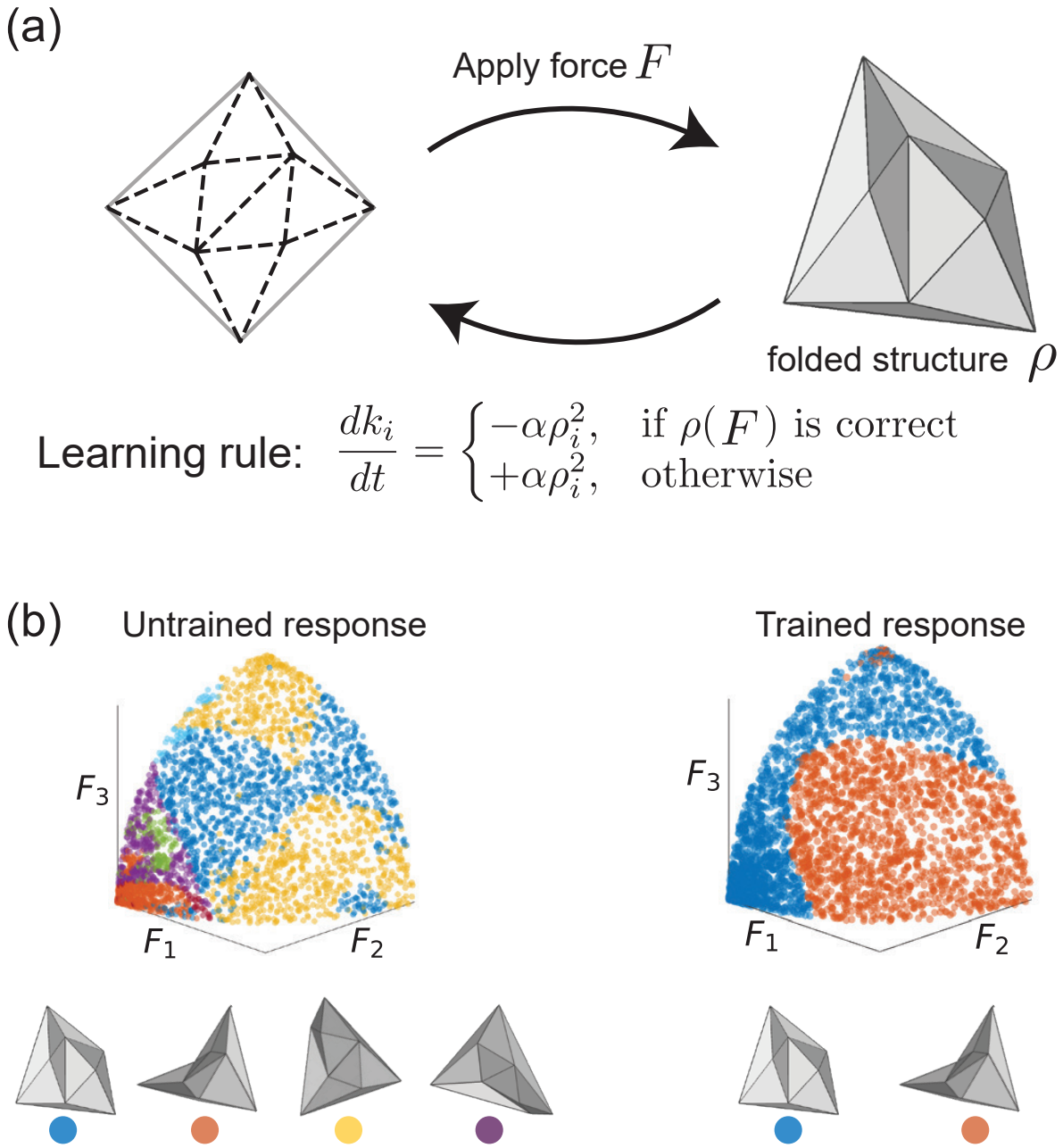

Figure 3.2: Supervised training of thin sheets. a) A sheet with random crease geometry is folded with a training force pattern F^a, resulting in a folded structure ρ. The stiffness k_i of each crease i is modified according to a local learning rule; if the folded structure ρ is the desired response for F^a as determined by a supervisor, creases soften in proportion to their folding strain ρ_i. Otherwise, creases stiffen. b) This rule trains the sheet to perform the desired classification of force patterns. The untrained sheet shows multiple folded structures in response to force patterns ($2d$ cross-section of force pattern space shown, with F_1, F_2, F_3 three particular directions in force space, see Supplementary note 1). The trained sheet shows only two folded responses that mimic the desired mapping of force patterns to folded structures.

A mechanical supervised training protocol

We pick 40 forces (20 blue, 20 orange) from the dog and cat distributions to serve as the training set, and apply our training algorithm described below to a thin sheet with uniformly stiff creases. In our training protocol, each of the training force patterns F is applied to the sheet in sequence, to obtain a folded structure $\rho(F)$. A supervisor determines whether the resulting folded state $\rho(F)$ is correct or incorrect by comparing the 3-dimensional geometry to that of a reference folded state ρ_{ref} for that class. (The reference state can be selected in several ways. Here, we average the response of the untrained sheet on training examples in each class; see Supplementary Note 2.)

We then apply the following learning rule that stiffens or softens each crease in proportion to folding in that crease,

$$\frac{dk_i}{dt} = \begin{cases} -\alpha\rho_i^r, & \text{if } \rho(F) \text{ is correct} \\ +\alpha\rho_i^r, & \text{otherwise} \end{cases} \tag{3.1}$$

for the stiffness k_i of each crease i. α is a *learning rate*, setting how fast stiffness values k_i are updated due to training examples (Here we use $\alpha = 10^{-4}$). r models non-linearities in strain-based softening or stiffening of materials; we use $r = 2$. Such plasticity is experimentally seen in several materials [21, 39, 32]; we discuss other learning rules and experimental constraints later. As we employ a physical model of origami sheets, we note that any rule that changes the stiffness of a crease has to be local, i.e. the stiffness of a crease i may not change due to the folding angle of a different crease j (which crease i has no information of). This is a major departure from learning rules in machine learning algorithms, which are typically non-local [4, 91]. For further information about this learning rule and its generalization to multi-class classification, see Supplementary Note 2.

After each round of training the pattern is unfolded back to the flat state. The same supervised learning step is then repeated in sequence for all training force patterns. A

training epoch is defined as one pass through the entire training set.

We find that as training proceeds, the number of observed folded structures decreases (fewer colors), and nearly all training force patterns fold the sheet into the 'blue' or the 'orange' labeled structures after epoch 40 (diamonds in Fig. 3.3b). The fraction of training force patterns that fold the sheet into the correct structure is defined as the *training accuracy*. This measure is an unbiased estimator of classification performance, as we choose the number of training forces to be the same in every class. Note that the different folded states of disordered sheets are typically highly distinct, even at the level of mountain-valley patterns [14, 109]; hence no special clustering algorithm is required to identify distinct structures and assign them distinct colors in Fig. 3.3.

However, a successfully trained sheet should correctly classify previously unseen 'test' force patterns, sampled from the same distributions. We tested the trained sheet by applying such test examples drawn from the distributions $S^{\mathrm{dog}}, S^{\mathrm{cat}}$ and recording the resulting folded structure (800 test force patterns for every class). In analogy to the training sets, the fraction of test examples yielding the correct folded structure is defined as the *test accuracy*. High test accuracy is observed (Fig. 3.3c,d) ($\sim$ 80% of the test examples classified correctly); thus the sheet generalizes and is able to have the right response to novel test force patterns through the changes induced by training examples.

Heterogeneous crease stiffness

Our learning rule facilitates classification by creating heterogeneous crease stiffness across the sheet (Fig. 3.4a). Indeed, as training proceeds, we find that the variance Δk^2 of stiffness grows (Fig. 3.4b, depicting the evolution of the stiffness profile for the learning process shown in Fig. 3.3). If the sheet is trained beyond the optimal point, the stiffness variance still grows, but the classifier eventually fails, as seen in Fig. 3.3b-c. The failure mode of over-training is typically that all forces fold the sheet into a single folded structure, resulting

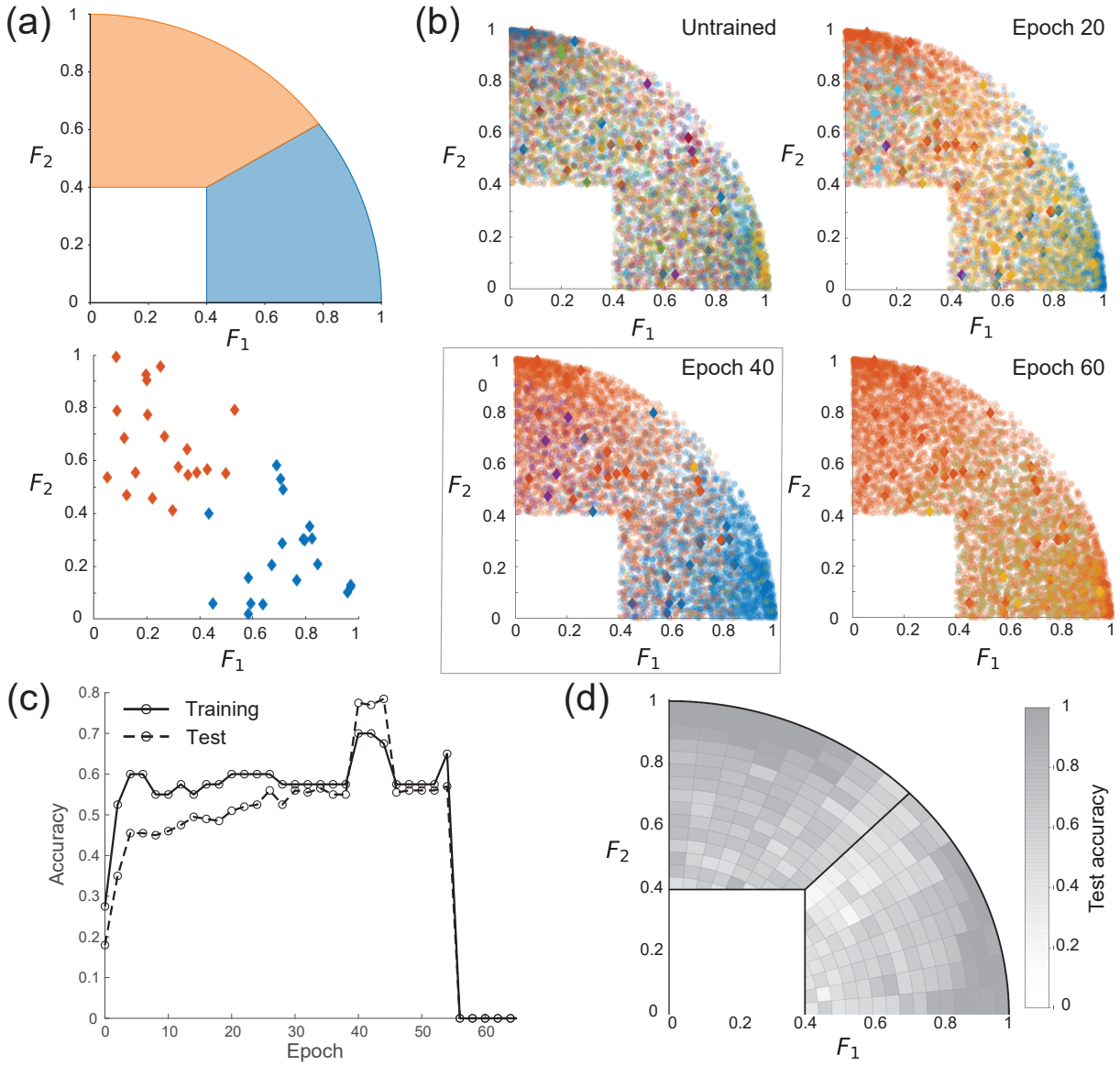

Figure 3.3: Supervised learning of cap-like force distributions. a) We define distributions S^{dog} (blue) and S^{dog} (orange) of force patterns in the space of applied forces (2d projection shown). 20 training examples (diamonds) are drawn from both distributions. b) An untrained sheet folds into many distinct folded structures (different colors) in response to applied force patterns. As training progresses, most force patterns are classified as either blue or orange according to the cap they belong to. When over-trained, all force patterns result in only one folded structure (orange). c) The trained sheet reaches peak performance after $\sim$ 40 epochs of supervised training (i.e., passes through the training examples). The trained sheet not only classifies the training set correctly (training accuracy), but generalizes to unseen test force patterns (test accuracy). d) The trained sheet is highly accurate when classifying force patterns near the center of the distributions, but less accurate close to the true decision boundary between the distributions.

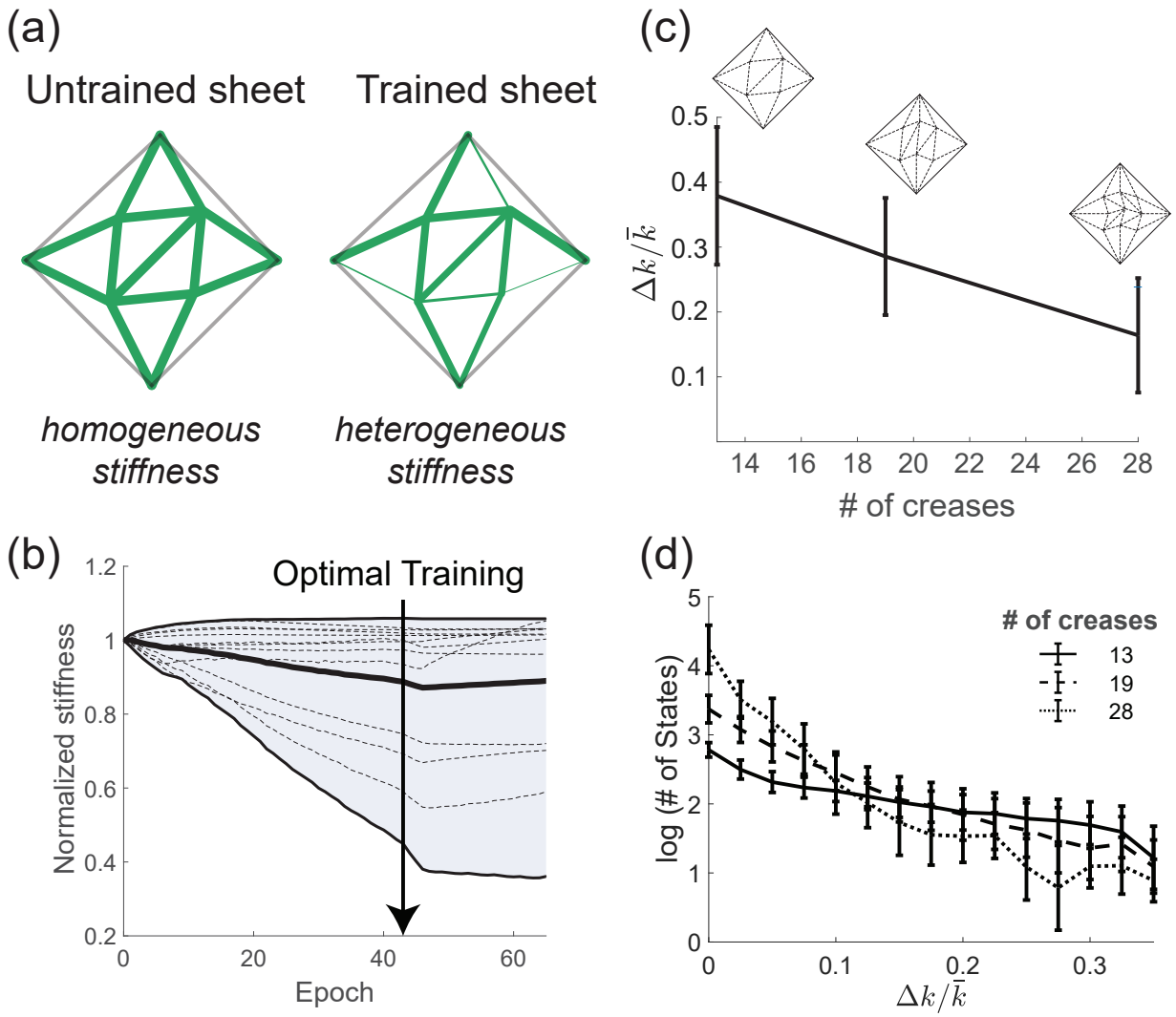

Figure 3.4: Training increases the variance of crease stiffness across the sheet. a) Untrained sheets have a homogeneous distribution of crease stiffnesses, while trained sheets have heterogeneous stiffness profiles (width of green lines). b) As the sheet is trained, the stiffness of different creases changes to different extents, such that the variance in stiffness values grows over training time (envelope shows the least and most stiff creases). c) Larger sheets with more creases require smaller variance in their stiffness values for optimal training. d) An untrained sheet starts with exponentially many available folded structures. During training, the number of available folded structures decreases exponentially with increasing stiffness variance Δk^2, allowing the sheet to classify a few distinct classes.

in no classification. This overtraining failure is associated with a large stiffness dynamic range, rather than with too small stiffness values.

We can understand this relationship between heterogeneous stiffness (Δk) and training using a simple model. A heterogeneous crease stiffness profile k with high stiffness k_i in crease i but no stiffness elsewhere, will lift the energy of structures ρ with small folding ρ_i in crease i less than structures with large ρ_i. Hence heterogeneous k can raise the energy of select structures, reducing their attractor size, while other structures remain low in energy and grow in attractor size. If we assume that folding angles ρ^a of structure a are randomly distributed (verified earlier in [86]) and assume a random stiffness pattern with standard deviation Δk, the energies $\sum_i k_i \rho_i^2$ of different structures will be distributed as a Gaussian with mean $\mu = \gamma \bar{k}$ and standard deviation $\sigma = \beta \Delta k$ where $\bar{k}$ is the mean stiffness, and γ, β some numerical parameters.

If structures above energy E_F are inaccessible to folding, the number of accessible folded structures is,

$$\#(\Delta k, N) \sim 2^N [1 - (\frac{\gamma \bar{k} - E_F}{\beta \Delta k})]. \tag{3.2}$$

Hence the number of surviving folded structures should decrease fast with Δk. This effect is indeed observed for trained origami sheets of different sizes (Fig. 3.4d). From numerical exploration of the energy distributions in this model, we find that γ is a constant regardless of sheet size, while $\beta \sim N^{-0.5}$ shrinks with sheet size (central limit theorem). Using this form of β in Eq. 3.2 predicts that the elimination of structures happens at a lower Δk for larger sheets, consistent with our results in Fig. 3.4c-d.

We conclude that as the training protocol proceeds, the stiffness variance Δk^2 grows, and the number of available folded structures decreases. The last surviving folded structures, reinforced by the learning rule of Eq (3.1), classify the force distributions correctly. Thus, the learning process merges attractors of the untrained sheet such that the surviving attractors

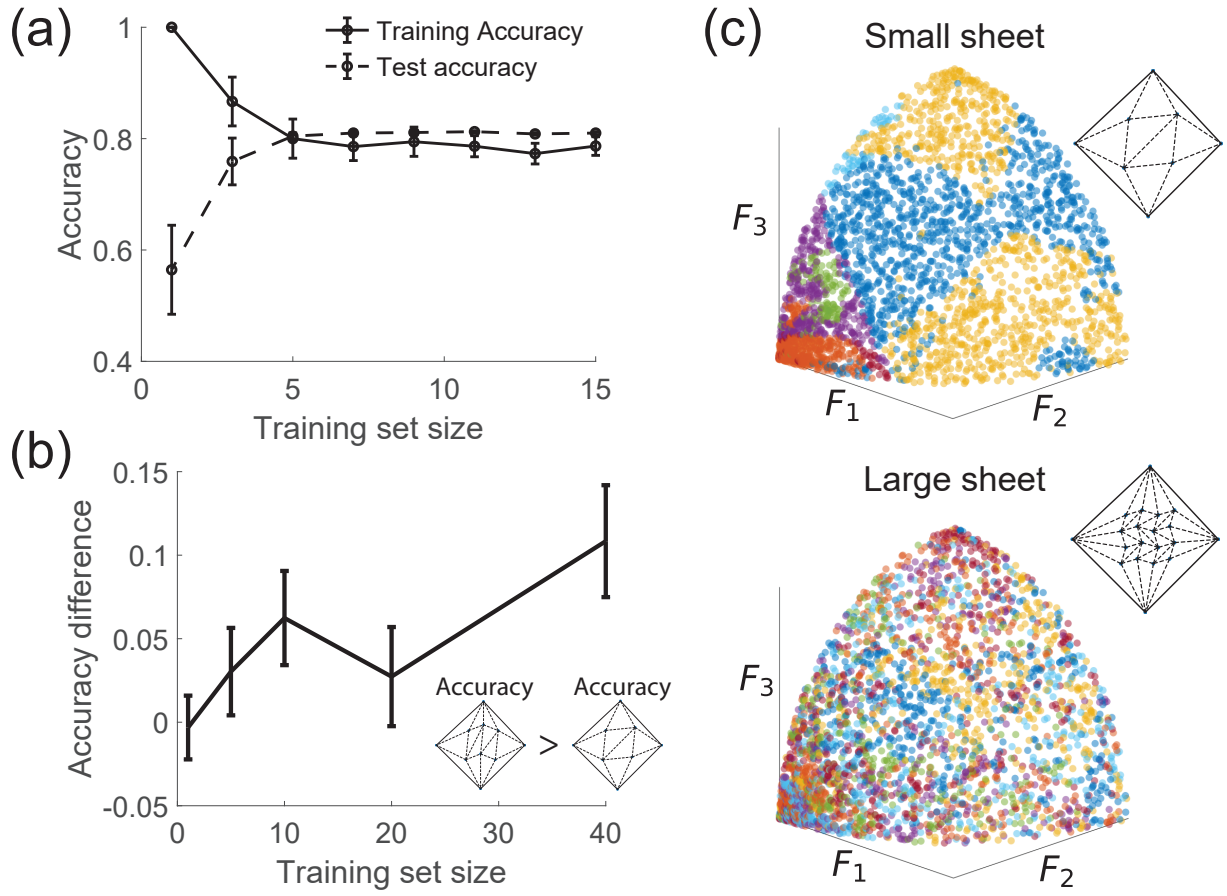

Figure 3.5: Effect of training set size and sheet size on test accuracy. a) With fewer distinct training examples, training accuracy is high but test accuracy is low (overfitting). Increasing the number of training examples improves test accuracy, at the expense of training accuracy. b) Sheets with more creases show larger improvements in test accuracy with increasing number training examples, as expected of complex models with more fitting parameters. c) A small untrained sheet (13 creases) shows ~ 10 folded structures (color coded) in response to different force patterns. A larger sheet (49 creases) sheet shows ~ 400 folded structures instead, each with smaller attractor regions in the space of force patterns. Consequently, larger sheets can create more flexible classification surfaces by combining smaller attractor regions; such complex models with more fitting parameters require more training examples to avoid overfitting.

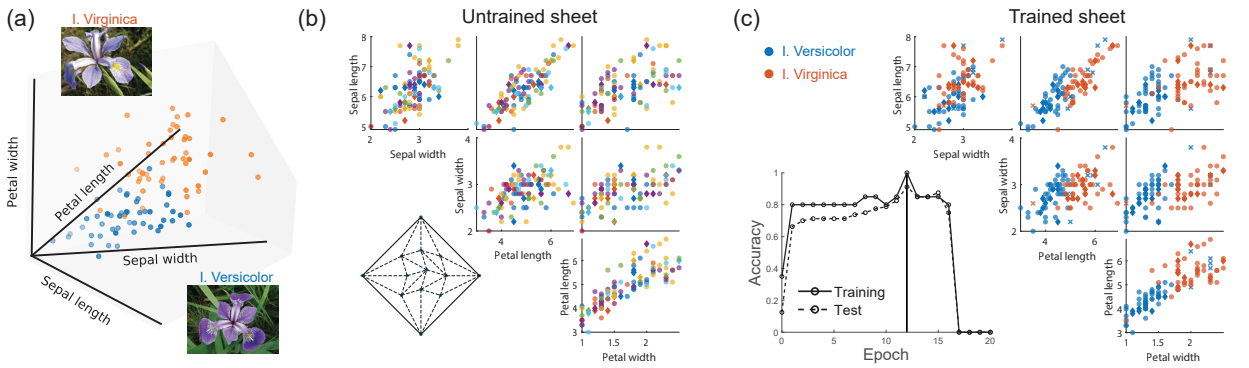

Figure 3.6: Training sheets to classify Iris specimens. a) We train a sheet to classify individual Iris specimens as one of two species based on petal and sepal lengths and widths [25]. We translate these four measurements into a spatial pattern of forces applied to the sheet. b) Folding response of an untrained (28 crease) sheet due to force patterns derived from the Iris data (shown in every cross section). c) The sheet is trained using 10 random examples (diamonds) of each species from the database [25] and then tested on 80 unseen test examples (circles). Matrix shows the classification of Iris flowers at optimal training of the sheet (91% test accuracy; mistakes denoted by x).

recapitulate features of the desired force-fold mapping.

Generalization and sheet size

Statistical learning theory [117] suggests that two critical parameters set the quality of learning: (1) the number of training examples seen, (2) complexity of the learning model. An increased number of training examples usually decreases training accuracy. However, test accuracy - i.e., the response to novel examples or the ability to generalize - improves. Furthermore, the improvement of test accuracy is larger for complex models with more fitting parameters. Intuitively, complex models 'overfit' details of small training sets, and thus show low test accuracy even if training accuracy is high.

Our sheets exhibit signatures of these learning theory results, with the size of the sheet (number of creases) playing the role of model complexity. For a sheet of fixed size, trained on the distributions of Fig. 3.3, we observe that increasing the number of training examples increases test accuracy and decreases training accuracy (Fig. 3.5a). In Fig. 3.5b, we find the

test accuracy of larger sheets with more creases improves more dramatically with the size of the training set, compared to smaller sheets.

These results suggest that sheets with more creases correspond to more complex classification models (e.g., a deeper neural network). For example, crease stiffnesses are the learnable parameters in our approach; hence increasing their number amounts to using a training model with more parameters. Further, untrained sheets with more creases support exponentially more folded structures [109, 14] as shown by the color coded force-to-folded-structure relationship in Fig. 3.5c. The training protocol achieves correct classification by merging different colored regions. Thus, larger sheets can approximate more complex decision boundaries by combining the smaller regions (Fig. 3.5c), and thus act as more complex models to be favored when the amount of training data is large. In the Discussion, we use these results to contrast memory and learning in mechanical systems.

Real-world classification problems

We have shown how disordered thin sheets can classify force distributions described by one feature (Fig. 3.3); one may ask whether these sheets can classify data described by multiple features.

We tested our learning protocol on the classic Iris data set [43], used extensively in the past to benchmark classification algorithms. This data set reports four measurements - length and width of petals and sepals - for individual specimens of different Iris species. While different Iris species cannot be distinguished by any one of these properties, we wanted to test if our sheet can learn the combination of features needed to distinguish species.

We picked the two most similar species in this data set, *Iris Versicolor* and *Iris Virginica* (Fig. 3.6a). We translated the four flower measurements to four force components applied to a sheet (see Supplementary Note 4). We then applied our training algorithm with a training set consisting of 10 examples of *I. Versicolor* and *I. Virginica* (diamonds in Fig. 3.6c). The

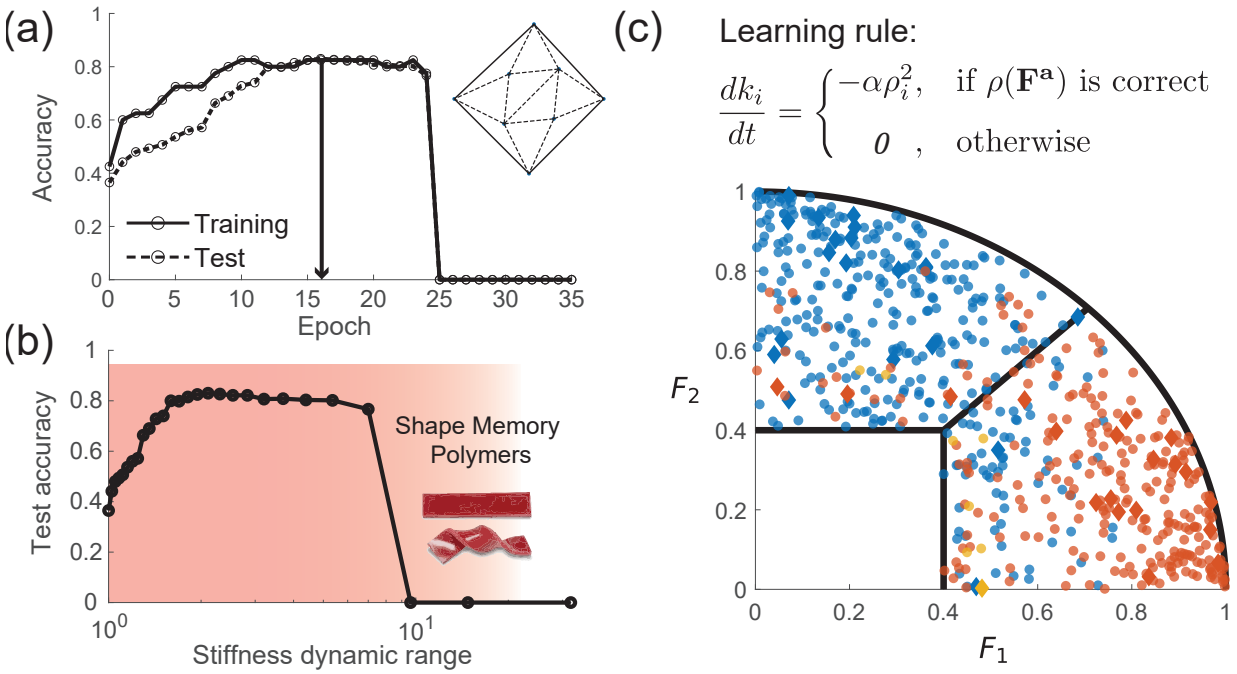

Figure 3.7: Learning is successful even with simplified training rules and experimentally realizable stiffness. a) A sheet (13 crease) trained on the classification problem of Fig. 3.3, with a simplified, experimentally viable learning rule shown in (c). b) At peak training, the dynamic range of crease stiffness values is ~ 2, well within the ranges supported by existing shape memory polymers (red region) [68]. c) Trained sheet reaches peak accuracy of 80% on test force examples (circles).

resulting trained sheet was tested on 80 *unseen* examples of these species; the trained sheet was able to identify the species of 91% of previously unseen specimens correctly.

We have tested our training protocol on more complex, higher dimensional distributions (Supplementary Note 3). For example, we used the folding behavior of one thin sheet (the master) as the target behavior for another thin sheet with a distinct crease geometry. We find that the trained sheet is able to correctly predict the response of the master sheet to forces not seen during training. Thus, using our training protocol, sheets can learn and generalize complex force-to-folded-response maps from examples.

Experimental considerations

Our learning framework requires materials that can plastically stiffen or soften when strained repeatedly [54]. Several such materials and structures are known, including shape memory polymers [68, 66], shape memory alloys (Nitinol) [19], and fluidic flexible matrix composites [85]. These systems have the advantage of truly variable, user controlled stiffness, and are used for various medical applications [9]. Poly-carbonate multi-layer sheets have been shown to produce controlled bending stiffness by more than an order of magnitude [36]. Other materials can show a plastic change in stiffness in response to aging under strain, such as Ethylene Vinyl Acetate (EVA) foam [44] and thermoplastic Polyurethane [12]. EVA was used recently [81] to show such behavior in a mechanical system trained for auxetic response. Another possible method, explored specifically for origami creases [46], controls the crease width (and hence stiffness) using photolitography [125]. We conclude that several available materials and experimental methods could implement our learning rule in an experimental setting.

The specific learning rule used in the paper requires the ability to soften or stiffen depending on the supervisor's judgement of outcome. Such a learning rule can be implemented by materials that stiffen under strain in one condition (say, high temperature, low pH) but soften under strain in another condition.

However, the results here also hold for simpler learning rules, e.g., that only require plastic softening under strain. For example, we can modify the learning rule (Eq. 3.1) to :

$$\frac{dk_i}{dt} = \begin{cases} -\alpha\rho_i^2, & \text{if } \rho(F) \text{ is correct} \\ 0, & \text{otherwise} \end{cases} \tag{3.3}$$

Such a rule is easily implemented with a strain-softening material with no stiffening needed. For example, if the folded outcome $\rho(F)$ is judged correct, we hold the sheet in the folded state $\rho(F)$ for a longer length of time (allowing significant softening) than when

the outcome is judged incorrect (no softening). We tested this simplified learning rule for the classification problem in Fig. 3.3; we find similar accuracy as earlier (see Fig. 3.7a). As discussed in supplementary note 2, the softening modification itself. e.g. the specific ρ^2 form, is not essential as well. Many similar monotonic $\dot{k} \sim f(\rho)$ would support learning in sheets.

Another significant experimental constraint is the dynamic range of crease stiffnesses k_i achievable in real materials without fracture at the creases. Fortunately, we find that for well trained sheets, the difference in crease stiffness is moderate (Fig. 3.4c), and does not exceed 30% of the mean stiffness value for a medium sized (28 crease) sheet. Fig. 3.7b shows that our required dynamic range in stiffness is ~ 2, well within the range for experimentally available materials such as shape memory polymers [68, 57]. Other materials like hydrogels and poly-carbonates exhibit bending stiffness that can easily adapt in a significantly larger range [36, 125], up to an order of magnitude.

Finally, another failure mode for our training protocol is overtraining. While the variance in crease stiffness Δk^2 is critical to eliminate attractors, overtraining can result in a sheet with only one folded state. Our analysis, presented in Fig. 3.4, suggests that large sheets should be easier to train experimentally since the stiffness variance needed is more moderate, while the transition to overtraining does not become much more rapid.

3.3 Discussion

In this work, we have demonstrated the supervised training of a mechanical system, a thin creased sheet, to classify input force patterns. As required for learning, the trained sheet not only shows the correct response for training forces, but can generalize and show the correct response to unseen test examples of forces. We studied the relationship between training error, test error, and the size of the sheet which plays the role of model complexity in supervised learning [117].

We can understand the promises and limitations of our *physical* learning approach better by considering similarities and differences with machine learning performed on *computers*, including for materials design [65, 103, 31, 33].

Generalizing by learning features: The core similarity is that both methods are predicated on learning 'features' from training examples (e.g., in Fig. 3.1d, something common to cats, yet not found in dogs) and thus classifying never-before-seen examples correctly.

In our context, 'features' are spatial correlations in forces applied to different locations on the sheet. For example, say, force patterns in class S^A might exert anti-correlated forces at sheet sites x and y while the force patterns in S^B show positive correlation at those sites. Meanwhile, all force patterns in S^A, S^B might exert forces at site z but those forces do not help distinguish S^A from S^B. The sheet must learn which combinations of forces at different locations to physically respond to (e.g., at x and y since they distinguish S^A vs S^B); and which combinations to ignore (e.g., at z). By learning such 'features', the sheet can *generalize*, i.e., respond correctly to unseen patterns with the same correlations.

In this way, learning can be contrasted with memory in mechanics [47]. A robust trained memory shows the correct response for all training examples (i.e., low training error), with no consideration of response to novel inputs. In contrast, we seek correct responses to unseen examples (i.e., low test error), even at some expense of training error. In this view, large sheets trained with limited training examples memorize while smaller sheets with more training examples generalize.

In spite of this similarity of our learning model to computational machine learning, it also differs significantly from machine learning algorithms in important ways:

Physical *in situ* training: The most significant difference is that our learning is not carried out on a computer but rather is the result of a natural physical process that changes crease stiffness in response to crease strain. That is, our sheet changes autonomously (adapts) to physical examples of desired behaviors, and consequently exhibits desired behaviors.

Materials with such intrinsic mechanical learning can be trained by an end-user, *in situ*, using examples of real forces relevant to the task at hand, rather than by a designer with a theoretical specification of use cases; such behaviors are sought e.g. in adaptive robotics [79]. Thus, we envision such physical learning as offering significant benefits compared to traditional machine learning and computational design in general.

Physical plausible local training: Physical plausibility of learning rules also constrain what can be learned; e.g., to maintain physical plausibility, we only explored a 'local' learning rule where stiffness of crease i is changed only by strain in crease i and not creases far away. Learning models in neuroscience are often restricted to be similarly local for biological plausibility (e.g., Hebb's rule). In contrast, artificial neural networks, trained on computers, face no such constraint; e.g., weights can be updated through back propagation to minimize a loss function, a highly non-local operation.

Our reliance on a physically plausible local process, instead of minimizing a loss function, might limit our capabilities compared to artificial neural networks. However, experience in neuroscience suggests that these limitations might be weaker than naively expected [60, 102]. For e.g., our local learning rules can still learn non-local correlations in input forces since the sheet's folding response to input forces is non-local, a property exploited by biological neural networks as well [4, 91].

Finally, unlike typical machine learning methods, our restriction to physically plausible learning rules that soften creases in proportion to strain allows for some interpretation for 'weights' (i.e., crease stiffnesses) learned. For example, if a force pattern results in an incorrectly folded structure, strongly folded creases due to this force will be stiffer than average after training. A more systematic understanding of the trade-offs between physical plausibility, interpretability and computational power of learning rules is critical to understand the limits of learned behavior in matter.

3.4 Supplementary Note

3.4.1 Supplementary Note 1 - Folding origami sheets

Energy of folded structures

The origami sheets used in this work are based on a self-folding origami energy model developed and validated in previous studies [5, 113, 86]. The effects of stiff creases are modeled by using torsional spring elements on each crease [109, 108]. Here we discuss in detail how the energy of a folded structure is computed.

For thin origami sheets with free-folding creases, the primary contribution to the energy of a folded structure is due to bending of the sheet faces. Instead of modelling the faces directly, we look at the mechanical constraints inherent to the geometry of the vertices. An origami vertex is known to apply 3 constraints on the dihedral folding angles of the creases connected to it (due to to embedding of the sheet in $3d$-space). The constraints can be derived by noting that the vertex must not tear open when folded. Thus, starting from any crease, alternating rotations about the dihedral and sector angles around the vertex have to result in an identity operation [113, 109, 108].

Suppose there are N creases denoted by an index i, each forded to an angle ρ_i, and N sectors with angles θ_i around the vertex. Rotations about one dihedral angle and one sector would combine to form a rotation matrix

$$R_i = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\rho_i & -\sin\rho_i \\ 0 & \sin\rho_i & \cos\rho_i \end{pmatrix} \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 \\ \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.4}$$

For the vertex to be closed (i.e. not torn open) in a folded structure, the combination of rotation about all crease dihedral angles and sector angles must be the identity:

$$A \equiv \prod_{i=1}^{n} R_i = I. \tag{3.5}$$

A folded structure with values ρ_i that do not satisfy Eq. 3.5 must cause the sheet faces to bend. Mathematically, this effect will manifest in finite off-diagonal values in the matrix $A \equiv \prod_{i=1}^{N} R_i$. As there are 3 independent non-diagonal elements, we say that the vertex imparts 3 mechanical constrains on the dihedral angles ρ_i around it.

At the flat state all $\rho_i = 0$ all constraints are trivially satisfied, so we can write down an expansion for the 3 off-diagonal terms of A ($T_1 \equiv A_{12}, T_2 \equiv A_{13}, T_3 \equiv A_{23}$) in powers of the folding angles:

$$T_a(\rho_i) = C_a^i \rho_i + D_a^{ij} \rho_i \rho_j + \dots \tag{3.6}$$

Then, the energy of breaking these constraints is taken as the sum of squares of the residues T_a of the constraint equations $E_{\text{Vertex}} \sim \sum_a T_a(\rho_i)^2$. Summing this vertex energy over all the vertices of the sheet gives rise to the total face bending energy. The energy due to folding of a stiff crease (modeled as a torsional spring with modulus κ_i) is quadratic in the folding angle $E_{\text{Crease},i} = \frac{1}{2} k_i \rho_i^2$. The total energy of a folded sheet with stiff creases is thus computed as

$$E_{\text{sheet}}(\rho) \equiv E_{\text{Face}} + E_{\text{Crease}} = \kappa \sum_{v \in vertices} \sum_{a=1}^{3} T_{va}(\rho_v)^2 + \frac{1}{2} \sum_{i \in \text{creases}} k_i \rho_i^2, \tag{3.7}$$

With κ the face bending stiffness scale (chosen as $\kappa = 1$ in this work), and k_i the creases stiffness values. The scale of creases stiffness is denoted by $\bar{k}$. The choice of stiffness energy scale plays an important role in our learning protocol. We have previously shown how the face bending energy scales like ρ^4 [108], while the crease stiffness energy scales like $\bar{k}\rho^2$. In turn, this gives rise to a transition folding angle scale in our model $\rho_c = \sqrt{\bar{k}}$. For large folding

angles $\rho \gg \rho_c$, sheet bending energy dominates, and the folding landscape is controlled solely by the sheet geometry. At small folding angles $\rho \ll \rho_c$ (close to the flat state), crease stiffness dominates, and it is possible to reshape the force-folding map. The goal of training is to reshape this map close to the flat state, such that the applied forces fold the sheet into desired folded states. Throughout this work, we choose an initial uniform crease stiffness $k_i = 0.02$. We find that trained sheets, though having heterogeneous stiffness profiles, still maintain a dominant stiffness scale at $\bar{k} \sim 0.02$. In our sheets the transition scale is thus given by $\rho_c \sim \sqrt{0.02} \sim 0.14rad$, a reasonable angle scale close to the flat state. To make learning in sheets feasible, we conclude that a stiffness scale $\bar{k} \sim 10^{-2}$ should be chosen.

The idea that heterogeneous stiffness at creases modifies the folding response of sheets is at the heart of our learning model. This approach was experimentally studied in lattice metamaterials, where stiffness heterogeneities in the form of negative stiffness cells are used to tune the material elastic properties [52, 29]. More recently, experimental studies have shown that heterogeneous stiffness can be used to avoid erroneous actuation pathways in metamaterials. Coulais et. al. have shown that a homogeneous stiffness hierarchical structure usually responds to actuation forces in disordered, undesired ways [17]. However, when the hierarchical metamaterial is designed rationally such that mechanical elements have different bending stiffness (different thickness), the structure is compactified in steps to obtain the desired final state. It is similarly known that self-folding with homogeneous crease stiffness usually folds incorrectly in response to folding forces [14, 109]. Biasing the creases to facilitate the correct folding can remove such undesired folding pathways so that the sheet folds correctly. Zhou et. al. considered designed heterogeneous thickness (and hence stiffness) of hydrogels via photolitography to control the buckling of sheets [125]. This method was used to remove unwanted pathways in origami, enabling robust folding of sheets into desired states [46].

Folding protocol

Now that the energy of every folded structure ρ_i of a specific sheet is defined. We can use this energy landscape to simulate the folding of the sheet. Experimentally there are multiple different ways to fold origami sheets [83], and we have previously outlined how these methods can be simulated numerically [108].

One way that an origami sheet can be folded is by applying torques directly to the different creases. Suppose a crease i of a flat sheet is subjected to an external torque F_i^{ext}. Such a torque will induce folding in the crease, but the sheet generally resists folding due to the extra energy that might be associated with a folded structure. Assuming that the folding process is over-damped, we may write a dynamical folding equation

$$\tau_{\text{relax}} \frac{d\rho_i}{dt} = -\frac{\partial E_{\text{sheet}}(\rho)}{\partial \rho_i} + F_i^{\text{ext}}, \tag{3.8}$$

where ρ is the current folded structure, and τ_{relax} a time scale of the over-damped dynamics. In this work we utilize a specific way of folding the origami sheets. Suppose a set of external torques F^{ext} is given (this could be a training or a test example as described in the main text). First, the sheet is folded very fast with a strong external torque F^{ext}, until a certain folding magnitude $\rho \equiv ||\rho||$ is reached. For fast folding we can initially disregard the sheet energy and thus get to a state

$$\rho_{\text{fast}} = \rho \frac{F^{ext}}{||F^{ext}||}.$$

Then the sheet is relaxed subject to the constraint that the overall folding magnitude is fixed (i.e. finding an energy minimum on a hyper-sphere of radius ρ in ρ-space):

$$\begin{aligned} \underset{\rho_i}{\text{minimize}} \quad & E_{\text{sheet}}(\rho) \\ \text{subject to} \quad & ||\rho|| = \rho. \end{aligned} \tag{3.9}$$

Finding a local minimum on the hyper-sphere guarantees that this folded structure would naturally occur if the sheet is folded with appropriate torques, as any neighboring configuration costs more energy, and the local minimum will attract the folding process. This algorithm is used to mimic experimental fast folding of origami sheets, followed by clamping of a crease at a specific folded dihedral angle. Here we also adjust the clamped angle such that the overall magnitude of folding ρ remains fixed and different (discrete) folded structures may be compared more easily. Such fast folding was tested extensively [108], and found to obtain the same results as numerically solving the ODE of Eq. 3.8.

Origami sheets and applied force patterns

In this project we use specific self-folding origami sheets. These are triangulated thin sheets, chosen to have the property of self-foldability. As discussed above, a single vertex induces 3 mechanical constrains on the angles of creases surrounding it. Thus each vertex has to connect at least 4 creases or it would be locally rigid. On top of that, for a sheet to self-fold, it needs to have one global degree of freedom, so that the number of creases needs to be one more than the number of constraints.

A simple way of generating patterns meeting these requirements is shown in Supplementary Fig. 3.8. These are 4 specific geometries used throughout this work as the sheets to be trained. Note that we label them according to their size, given by the number creases in each sheet. The number of creases in these sheets are $13, 19, 28, 49$ and the numbers of internal vertices are $4, 6, 9, 16$. Subtracting 3 times the number of vertices from the number of creases leaves us with one global degree of freedom for each of these sheets, as required.

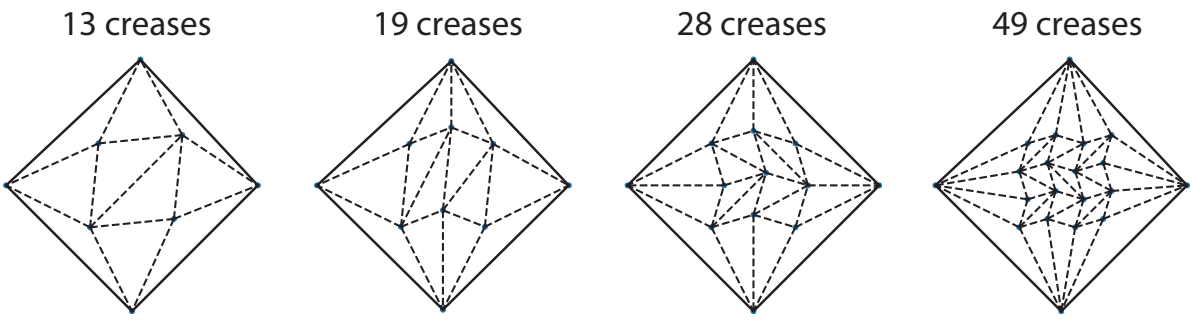

Figure 3.8: Origami Sheets used for training. The size of each sheet is determined by the number creases.

The number of supported folded structures for these sheets grows exponentially with the number of internal vertices, such that these sheets can fold in approximately $2^4, 2^6, 2^9, 2^{16}$ distinct ways [109, 14]. In fact, any sheet with these topologies (yet different geometries) will have a similar number of distinct folded structures. The exact details of the supported folded structures is dependent on the specific geometry, but we only require the existence of many distinct folded structures for the purpose of training.

These specific sheets, used for training classifiers throughout this work, are definitely not special. We attempted training classifiers using sheets with different geometries and obtained comparable results. In analogy to learning algorithms, the details of the sheet and its supported folded structures correspond to the family of models that the training protocol selects from. For origami, we believe the available classification models are given by merger of attractors of folded structures, supported by the sheet. Since the number of available models to choose from is exponentially large, we reason that the geometry of the sheet should play little role in the success of classification. Therefore, any self-folding origami sheet could be used for training classifiers.

The choice of force patterns applied to the sheets is constrained by the problem definition as training and tests sets. Still, there is usually freedom in how these forces are applied. For example, suppose we wish to train the 13 crease sheet of Supplementary Fig. 3.8 on $2d$ force distributions, such as the spherical caps shown in Fig. 3 in the main text. The training and

test sets could thus be supplied as pairs of numbers, together with a label (blue\orange). A simple choice for training on such a data set is to pick two creases in the sheet and apply torques directly to these creases, as in Eq. (3.8). Here we utilize a different approach.

For an untrained sheet with homogeneous stiffness, it is known that all folded structures reside in the linear null space of the vertex constraint matrix C at the flat state [109]. Thus, forces applied in a direction within this null space are more 'natural' for the sheet, and in general cost much less energy due to face bending. We compute the span of the null space for each one of these sheets, and find that the dimension of the null space is $d_{NS} = \#_{\text{creases}} - 2\#_{\text{vertices}}$. Therefore the 13 crease sheet has a $5d$ null space, while the 49 crease sheets has $17d$ null space. Then, the training and test examples are mapped to forces in the null space as follows. For a $n-d$ data set, we choose n random orthonormal vectors in the null space. Each training\test example is mapped to a force pattern by assigning every component to one of the random orthonoraml vectors. Now these forces can be directly applied to the sheet to facilitate the training protocol.

Before training, we choose the stiffness values to be uniform. This choice is deliberate, as it allows the training protocol to access the entire set of supported folded structures. We find that initializing the stiffness elements with a substantial poorly chosen heterogeneous profile negatively affects learning. This is expected, as poor initialization can completely eliminate good folded states which could be useful for classification. The training protocol results in *learned* heterogeneous crease stiffness that facilitates the correct classification. We observe that a heterogeneous stiffness changes the geometry of the folded structures, so that they do not strictly reside in the null space of the untrained sheet. Still, for the moderate heterogeneity developed during training, the folded structures are very close to the null space, such that the described mapping is still useful and practical.

3.4.2 Supplementary Note 2 - Training origami sheets

Learning rule

As discussed in the main text, self-folding origami sheets naturally give rise to complex mapping of force patterns to folded structures, with exponentially many structures supported by the sheet. The learning rule developed in this work is meant to modify that map by changing crease stiffness coefficients, such that only a small number of folded structures are retained, corresponding to the desired classes. Here we will define precisely how the learning rule is chosen and applied to the sheet in order to develop the desired mapping.

According to the specification of the classification problem, the trainer has no a-priori knowledge of the true underlying force distributions. Instead they are supplied with a list of labeled force patterns ('cats' and 'dogs'). These training examples are used to find a reference folded structure in the following way. We fold an untrained sheet with every 'dog' example in the training set and record the folding angles of the obtained folded structures. Then, a reference 'dog' structure $\hat{\rho}_{\text{dog}}$ is defined as the average of all of these folded structures (normalized appropriately)

$$\hat{\rho}_{\text{dog}} \equiv \frac{\sum_{F\in\mathcal{F}^{\text{dog}}} \rho_U(F)}{||\sum_{F\in\mathcal{F}^{\text{dog}}} \rho_U(F)||}, \tag{3.10}$$

with $\mathcal{F}^{\text{dog}}$ the set of 'dog' training force patterns and $\rho_U(F)$ the folded response of the untrained sheet to force pattern F. A similar reference state $\hat{\rho}_{\text{cat}}$ is obtained for the 'cat' training examples. Crucially, once the reference structures are set for the untrained sheet, they are kept fixed throughout the training process. These reference structures are used to define the learning rule discussed in the main text. Suppose that during the training protocol, we choose a random 'dog' example F^{dog} and apply it to the sheet. The normalized resulting folded structure is written as $\rho(F^{\text{dog}})$. The learning rule then compares this folded

structure to the reference structures defined above and the stiffness coefficients are modified as follows:

$$\begin{aligned} &\text{if } \rho(F^{\text{dog}}) \cdot \hat{\rho}_{\text{dog}} > \rho(F^{\text{dog}}) \cdot \hat{\rho}_{\text{cat}} : \quad \frac{dk_i}{dt} = -\alpha \rho_i^r(F^{\text{dog}}) \\ &\text{else} \ : \qquad\qquad\qquad\qquad\quad \frac{dk_i}{dt} = +\alpha \rho_i^r(F^{\text{dog}}), \\ &k_i \geq 0, \ i \in \text{creases} \end{aligned} \tag{3.11}$$

where we choose $r = 2$. In essence, the learning rule checks whether the observed folded structure is closer to the 'dog' reference then to the 'cat' reference. If it does, the stiffness of creases that fold considerably in that structure is reduced, effectively reinforcing this force-fold mapping. An opposite modification occurs if the folded structure is far away from the 'dog' reference. A similar training rule is used when 'cat' forces patterns are applied, with the understanding that we wish to compare the resulting folded structure $\rho(F^{\text{cat}})$ to the 'cat' reference $\hat{\rho}_{\text{cat}}$. The intuition for this learning rule is that the softer a crease is, the more it will tend to fold. Thus if the sheet responds to a force in a desired way, making the creases that fold more softer will increase the likelihood that it will continue acting in the right way when subject to that force. Conversely, if the sheet does not respond correctly, stiffening reduces the likelihood it will respond incorrectly to that force in the future. This intuition sheds light on the question of inseparability in our model, as creases that correlate with certain features in the classified data will tend to be softer after training.

As discussed in the main text, our learning modifies stiffness according to the strain energy at each crease $\Delta k \sim \rho^r$, with $r = 2$. We have considered training sheets with other values of r in the range $1 - 5$, as seen in Supplementary Figure 3.9. These trials gave rise to qualitatively similar results. We thus elected to use $r = 2$ for our classification problems.

Note that while our learning rule is local in the space of creases, it can still learn non-local correlations in the space of input forces. It is believed that biological systems use local

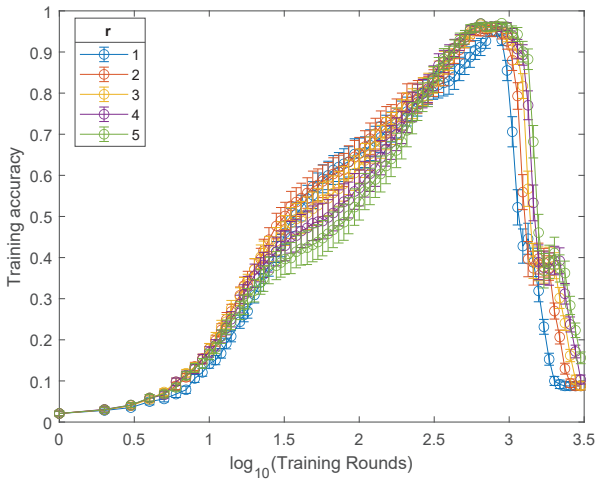

Figure 3.9: Training sheets to fold as desired with different values of the power parameter r in Eq. 3.11. We observe small differences in the accuracy obtained using different values of r. Throughout this work we use $r = 2$, an experimentally viable choice.

learning rule [91], an idea often stated as 'Hebbian learning' [35]. Though such local learning rules are in principle less powerful than arbitrary non-local rules, they can indeed facilitate learning in complex data sets [72, 4].

This learning rule can naturally be generalized to more than two classes. If c classes are to be classified, one could define c reference folded states. Then Eq. 3.11 could be used for learning from given training examples, with a simple modification; Crease i should be softened in proportion to the folding angle ρ_i^2 if the folded state is closer to the appropriate reference state. Otherwise, the crease should be stiffened.

Assigning labels to folded structures

To begin with, we are given labeled force patterns, and an untrained sheet with many available folded structures. It is important to note that these folded structures are equivalent and not intrinsically labeled. Thus, as part of the learning protocol we must specify how to label these folded structures, and in particular which of them to call 'dog' and 'cat' (or 'blue' and 'orange'). A simple solution would be to choose 2 of the folded structures in

advance and assign the classification labels to them. Unfortunately, this turns out to be too restrictive for a couple of reasons. First, the choice may be far from ideal in the sense that these labeled folded structures are very different than the actual folded response of the sheet to the labeled force patterns. Furthermore, as the training process modifies the stiffness of different creases, the folded structures supported by the sheet change as well, either by moving around or disappearing altogether in saddle-node bifurcations [108]. We thus take a different approach to labeling folded structures, as detailed below.

Suppose we have trained a sheet for some time, and it now has a particular stiffness profile on its creases k_i. To find a folded structure of this sheet to be labeled 'dog', we apply each of the 'dog' training examples once, and record the discrete resulting folded structures due to all of them $\{\rho(F \in \mathcal{F}^{\text{dog}})\}$. We then count the training force patterns that folded into each one of the structures in this set. The folded structure that resulted from the largest number of training force patterns is chosen to be labeled as 'dog'. In case of a tie, e.g. two or more folded structures folding as a result of the same number of force patterns, one of these structures is randomly chosen to serve as the label. Thus, the labels for 'dog' and 'cat' are decided through simple plurality rules every time we compute the classification accuracy of the sheet. Note that force patterns may also fold the sheet into structures not labeled as either 'cat' or 'dog', in which case they count as failed classification. If both 'dog' and 'cat' labels are chosen to be associated with the same folded structure, a plurality rule between the two classes decides which class is labeled with that structure (i.e. whether more 'cat' or 'dog' force patterns folded into that structure), while the other is assigned with the runner up structure of that labels' plurality vote. Finally, if the sheet is over-trained to the point where only one folded structure remains, that structure is labeled as both 'cat' and 'dog', such that classification fails completely, by definition.

Effective cost function

In this work we have defined our learning rule as a supervised physical process modifying the stiffness coefficients of an origami sheet. It is interesting to compare this kind of learning protocol to more established learning algorithms originating in computer science and statistics. An important difference is that traditional learning algorithms are usually defined as an optimization problem, where the function to be optimized (often called cost or loss function) incorporates the training data.

A simple example of a learning algorithm is linear regression, where the cost function is usually chosen as a least squares form, with differences taken between a linear model $h(x)$ and the observations y:

$$\begin{aligned} \text{Cost} &\equiv \sum_{d \in \text{data}} \left(h(x_d) - y_d\right)^2 \\ h(x) &= a_0 + a_1 x \end{aligned} \quad . \tag{3.12}$$

The regression (or learning algorithm) then optimizes the cost function with respect to the model parameters $a \equiv (a_0, a_1)$

$$\underset{a}{\text{minimize}} \quad \text{Cost}(\{x\}, \{y\}; a).$$

This optimization can be performed in any number of ways, but a practically favored method (at least for more advanced algorithms like deep learning) is mini-batch stochastic gradient descent (SGD) [99]. In an extreme case, when the mini-batches are chosen to be of size 1, a single training example (x, y) is chosen at random in each step, and one computes the gradient (with respect to parameters a) of the cost function defined with this example alone $G \equiv \nabla_a (h(x) - y)^2$. Now, training proceeds by modifying the parameters in proportion to the the gradient of this single example cost function

$$a \to a - \alpha G, \tag{3.13}$$

where α is a scalar known as the learning rate. We may compare this single example SGD with our origami training protocol. It is relatively easy to see that our training rule (Eq. 3.11), once a standard wait time is chosen at the folded state, has the form of SGD, making it similar in essence to other learning algorithms. To find out what effective cost function gives rise to the origami learning rule, we integrate Eq. 3.11 with respect to the stiffness coefficients

$$\begin{aligned} &\text{cost}_{\text{map}}(\rho(F^{\text{dog}})) = f \sum_{i \in \text{creases}} k_i \rho_i^2(F^{\text{dog}}) \\ &\text{if } \rho(F^{\text{dog}}) \cdot \hat{\rho}_{\text{dog}} > \rho(F^{\text{dog}}) \cdot \hat{\rho}_{\text{cat}} : \qquad f = +1 \\ &\text{else} \ : \qquad\qquad\qquad\qquad\qquad\quad f = -1 \end{aligned} \quad . \tag{3.14}$$

Similarly to the linear regression example, our origami training protocol attempts to minimize this derived cost function, one training example at a time. Inspecting this function, note that it is very similar to the energy of the torsional springs in the folded structure $E_{\text{Crease}}(\rho) \sim \sum_i k_i \rho_i^2$. The difference is in the 'supervising factor' f that can be ± 1 whether the folded structure is accepted or not. We conclude that our origami training protocol is attempting to minimize the energy of accepted folded structures, while maximizing the energy of rejected structures. It is however important to note that the origami model does not have a fundamental cost function to optimize, but instead a local learning rule, from which a cost function emerges.

Complexity of origami classification

Self folding origami is often associated with difficult (NP-complete) computational problems. The Classic work of Bern and Hayes has shown that determining whether a sheet is rigidly foldable is NP-complete [6]. More recently, it was shown that even folding a given sheet to a desired folded state is NP-complete [109]. These NP results apply to a sheet with soft creases; consequently there are many ways (e.g., MV assignments) of incorrectly folding the sheet. In fact, the result can be intuitively understood by a mapping from folding origami to a satisfiability (SAT) problem of a set of equations (one for each vertex) with boolean variables representing the M or V state of each crease at that vertex. Such a SAT problem can also be visualized as a spin glass Hamiltonian with many local minima [3].

Thus, we expect that no efficient (polynomial time) algorithm can modify *all* origami sheets in a way that makes them easy to fold. Fortunately, computational complexity is a statement about the most difficult instances of a particular problem. It is certainly possible that an efficient algorithm can fold a *typical* sheet in a desired way. We have previously described such an algorithm [108], based on linear or quadratic programming, that selects the correct crease stiffness on creases to support easy folding of the sheet. It was shown that this algorithm can facilitate easy folding of many sheets, which would otherwise require great care to fold correctly. This idea, that the right crease stiffness heterogeneity can be used to make a typical sheet fold in desired ways, is used as the basis of our learning algorithm presented in this work. When crease stiffnesses are introduced, many or all of these incorrect ways of folding can be made energetically unfavorable. In the SAT or spin glass analogy, stiffnesses can be viewed as fields or biases for the variables that lift many of the minima. Thus, the stiffnesses found by our algorithm modify the relevant SAT problem until it is easily solvable. Finally, we must note that results about NP-hardness are worst case results; i.e., there exists at least one sub-class of problems that are exponential time to solve. The supervised learning framework here works with reasonable consistency but any

such statistical approach that typically works cannot contradict any NP-hardness results, since it fails on some problems.

Supervised learning, and supervised classification specifically, are NP-complete problems as well [11]. Given a large set of constraints (data points) and a certain family of models, we do not know of an efficient algorithm guaranteeing a set accuracy of classification. In this way, machine learning is similar to problems in physics such as spin glasses [3], and self folding origami. With regards to complexity, the learning protocol suggested in this work is similar to machine learning algorithms. Neither our learning rule nor traditional algorithms guarantee an accurate classification for a specific data set and model (in our case, specific sheet). However, it is known experimentally that an accurate solution to a classification problem can be found by using more expressive models (e.g. a deeper neural network). Similarly, we find that larger sheets with more stiff creases provide better classification results, as shown in the main text.

Example classification problem

In this section we provide further detail about the example classification problem discussed at length in the main text, and shown in Figure 3. While it is not a standard benchmark classification problem, we wish to include this extra information here for the sake of future reproduction of these results by other physically motivated learning models. The full data set, including the training forces and the progress of the training protocol, as well as MATLAB codes for training the sheet, are included as supplementary files.

As described in the main text, we use a 13 crease sheet to classify forces drawn from two classes (Figure 3a). The initial sheet has uniform stiffness on all creases ($k_i = 0.02$, in the units where bending stiffness is chosen as 1). We consider the $5d$ null-space of the sheet and classify forces in that space, Force directions F_1 and F_2, defining the distribution to be classified, are two random orthonormal directions in this null-space. We draw 20 training

forces from each class, dog and cat, given according to the distribution $S^{\text{dog}} = \{F | F \cdot F_{\text{dog}} \geq D, F \cdot F_1 > F \cdot F_2\}$, and similarly for S^{cat} for a threshold $D = 0.6$. The forces we pick are normalized, so that they live on the surface of a $5d$ sphere, but only 2 of these dimensions are relevant for classification. Therefore, if we sampled forces with small component in the $F_1 - F_2$ plane, they would be extremely hard to classify. For this reason we choose the cutoff $D = 0.6$, ensuring the sampled forces have a significant component in the relevant space.

Once the training forces are picked, we also sample 800 test forces for each class from the same distribution. The training and test forces are randomly ordered. We note that while the order of training examples affects learning, these effects do not change results qualitatively, as long as all training examples are shown. To train the sheet, we go through the training examples, alternating the class at every iteration. We fold the sheet with these training forces and apply the update rule of Eq. 3.11 given the obtained folded state. We choose the learning rate $\alpha = 10^{-4}$. After exhausting all of the training examples, we say the training has advanced by one epoch. Then, we continue training on the same training set for as long as necessary. Data for this simulation, as well as MATLAB codes for training the sheet, are available as supplements.

3.4.3 Supplementary Note 3 - Using origami sheets to define classification problems

The force distributions classified in the main text are relatively simple. Both the spherical cap and the Iris data distributions can be well separated by a hyper-plane, a very simple decision boundary. It is interesting to study the type of decision boundaries naturally trainable in origami sheets – and whether they can be used to classify intrinsically high dimensional data.

There are many ways to obtain high dimensional distributions. Here we choose to study distributions derived from the folding maps of origami sheets. Consider a relatively simple sheet with 2 internal vertices (Supplementary Fig. 3.10a). It is known that such sheets sup-

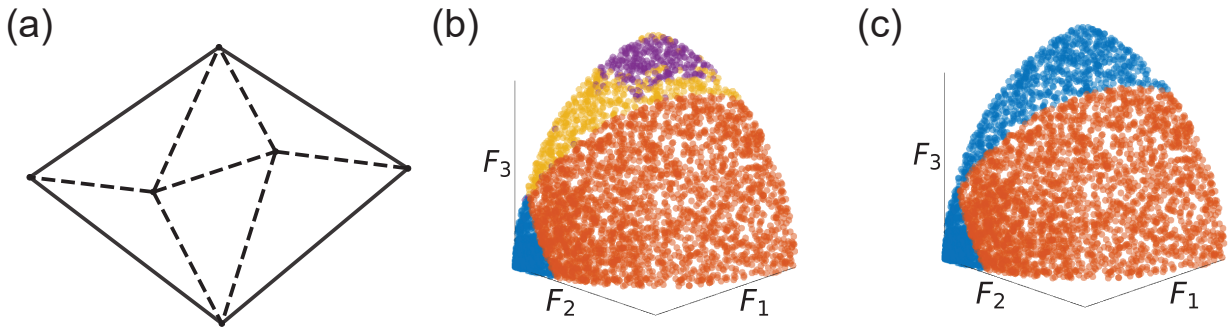

Figure 3.10: Defining force distributions using the force-fold mapping of an origami sheet. a) Origami sheets with 2 internal vertices support 4 discrete folded structures. b) Sample force patterns on a 2-sphere show the force-fold mapping (4 color coded regions). c) When some attractor regions are merged (here, blue, yellow and purple are merged), we obtain an intrinsically 2-dimensional separator surface between two classes of force patterns.

port 4 discrete folded structures, and that the linearized null space in which they reside is 3-dimensional. Therefore, if we sample random force patterns within this null space, we expect to see the sheet folding into 4 distinct structures (color coded regions in Supplementary Fig. 3.10b). The forces F_1, F_2, F_3 are assigned by randomly choosing Euler angles on the 2-sphere, and 3000 data points are sampled on the positive octant. Note that we sample normalized forces on the surface of a $2-sphere$, such that the distribution of force patterns is actually 2-dimensional.

Now, suppose we wish to classify forces to 2 classes ('blue'\'orange'). A simple way to create 2 neighboring sets of points is to take the data of Supplementary Fig. 3.10b and merge some attractor regions to create larger groups of points. In Supplementary Fig. 3.10c, we merge the 'blue', 'yellow', and 'purple' folded structures to create one region we define as 'blue'. This process yields two distributions that are intrinsically 2-dimensional, and not naturally separable by a hyper-plane. Larger sheets can be similarly used to create force distributions in higher dimensional space.

With this process, we have access to a new variety of 2-way classification problems, on which we can try to train origami sheets using the training protocol described in the main

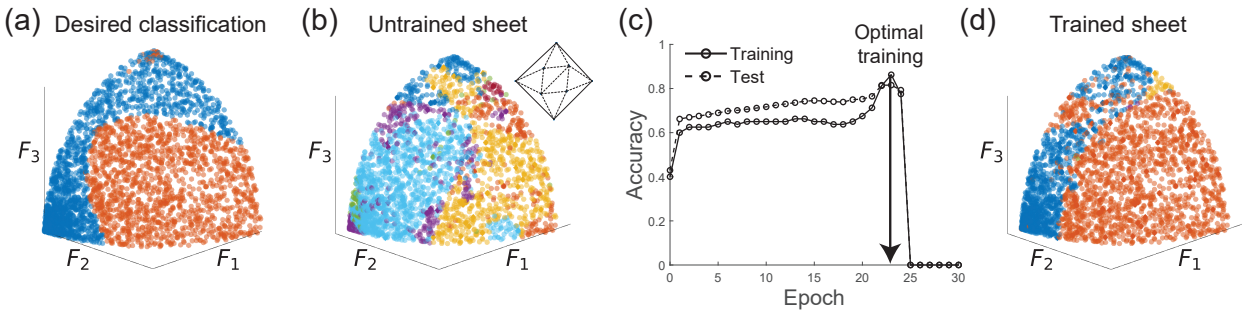

Figure 3.11: Training a sheet on a force distribution derived from a different sheet. a) Target classification, a sample distribution derived from a small, 2-vertex origami sheet. b) The force-fold map of an untrained 13 crease sheet is very different from the desired mapping. c) With training, the accuracy of classification improves and peaks at 82%. d) The optimally trained sheet has a complex decision boundary that resembles (but different than) the desired boundary.

text. Crucially, the sheet used to classify such distributions is different than the sheet used to derive the distribution. In other words, we ask if our training protocol can induce an origami sheet to mimic the force-fold mapping of another sheet.

Suppose we want to classify the distribution seen in Supplementary Fig. 3.11a, derived form a 2-vertex sheet as described above. We wish to train a 13 crease sheet to classify this force pattern data. The untrained sheet has 2^4 discrete folded structures that do not align with the target distribution in any representation that we tested (Supplementary Fig. 3.11b). The problem of classification here is to train this sheet to have just 2 folded structures with the right force-fold mapping as in the target distribution.

The target distribution is mapped to applied force patterns on the 13 crease sheet by the construction describe in Supplementary Note 1: choosing random orthonormal vectors in the null space of the 13 crease sheet and mapping the distribution as components of these vectors. We then randomly sample 20 'blue' and 20 'orange' force patterns, marked as diamonds in Supplementary Fig. 3.11, to serve as the training set. As we train the sheet, the classification accuracy improves dramatically and reaches a maximum of 82% (test accuracy) after 23 epochs (Supplementary Fig. 3.11c). To qualify the classification better, we look at the classification results corresponding to the maximal accuracy at epoch 23 (Supplementary

Fig. 3.11d). We observe that the trained decision boundary resembles the desired boundary, so that the training protocol indeed produced a reasonable classification.

Note a few artifacts that still remain in the trained map: 1) there are 3 folded structures left, rather than 2 (a small third color coded region exists, labeled yellow), 2) a second orange region appeared inside the bulk blue region, emphasizing that the decision boundaries between folded structures in sheets are generally *not* hyper-planes. We conclude that origami sheets can be trained to classify distributions derived from other sheets, that are intrinsically higher dimensional than the problems discussed in the main text. Moreover, the decision boundaries are non-linear, so that in principal sheets can classify data that is not linearly separable. We leave questions of the sheet size and the complexity of decision boundaries to future studies.

3.4.4 Supplementary Note 4 - Transforming Iris data to applied forces on sheets

The Iris data set [25] classified in the main text is a classical problem for classification. In this work we are able train an origami sheet to correctly classify two species of Iris (I. Versicolor, I. Virginica) at an accuracy of 91%. Here we discuss how the Iris data is used to generate training and test sets of applied force patterns to be used on origami sheets.

Each Iris example in the data set is given as a vector with 4 features (components): sepal length, sepal width, petal length, petal width. These length measurements are all given in *cm*. In addition to these measurements, each Iris is labeled as one of the Iris species in the study. To generate force pattern sets from this data, we would like the different measurements for each Iris example to be components of force vectors in the null space of the origami sheet, as described in Supplementary Note 1. However, the raw Iris data is not suited for this purpose due to two reasons. The dimensionful measurements of Iris lengths, if directly translated to forces, would be far too great for our sheets and will cause it to

fold too much and cause the sheet faces to collide. More crucially, sepal and petal lengths tend to be considerably larger than their widths, and the same goes for the variance of these variables. This will causes the width variables to be perceived as less important in the training protocol, and have a negative effect on the classification results.

Fortunately, diverse data like this is an issue regularly faced by learning algorithms, and it is generically solved by applying an invertible transformation to the data. The transformed data is then better suited for the learning algorithm in use. A typical example of such a transformation in data sets is to normalize each feature (divide by the mean of that feature) and translate it such that the mean of the transformed data is 0. This transformation is especially useful for classification algorithms like logistic regression, where the different features have different dimensional units.

In our case however, the standard transformation above is not useful, due to a particular property of origami sheets, namely their Z_2 symmetry. If forces F are applied to the sheet and it folds into a state ρ, then folding the same sheet with forces $-F$ will result in a state $-\rho$. This is true for any self-folding origami sheet, regardless of the stiffness profile on its creases. This property cannot be changed by training the sheet. Thus, force patterns of opposite sign and different labels cannot be correctly classified. A simple way to avoid this issue is to limit the force patterns to reside in a restricted part of force space. We choose to limit the distributions such that the transformed Iris data will all be in the positive 4-hyperoctant.

In addition, we want the data to span as much as possible of the positive hyperoctant. This will increase the expressive of our training protocol, as more discrete folded structures would become available if the applied force patterns are more diverse. We thus need to transform the Iris data to be all positive, and stretch it such that all features have similar variance.

To achieve these goals we apply the following linear (invertible) transformation to the Iris data of the Versicolor and Virginica species. Suppose an Iris example is given as a vector

x (where the components are sepal length, sepal width, petal length, petal width in this order). The vector is transformed by

$$x^* = Ax + b$$

$$A = \begin{pmatrix} 0.264 & 0 & 0 & 0 \\ 0 & 0.580 & 0 & 0 \\ 0 & 0 & 0.303 & 0 \\ 0 & 0 & 0 & 0.836 \end{pmatrix}, \quad b = -0.880. \tag{3.15}$$

Then the transformed vector is used to define the force patterns applied to the origami sheet, as described in Supplementary Note 1. After training is concluded, the transformation can be inverted to relate the origami classification results with the original Iris data, as shown in the main text.

REFERENCES

[1] Andrew Adamatzky. *Physarum machines: computers from slime mould*, volume 74. World Scientific, 2010.

[2] Jan A Åström, PB Sunil Kumar, Ilpo Vattulainen, and Mikko Karttunen. Strain hardening, avalanches, and strain softening in dense cross-linked actin networks. *Physical Review E*, 77(5):051913, 2008.

[3] Marco Baity-Jesi, Levent Sagun, Mario Geiger, Stefano Spigler, Gérard Ben Arous, Chiara Cammarota, Yann LeCun, Matthieu Wyart, and Giulio Biroli. Comparing dynamics: Deep neural networks versus glassy systems. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124013, 2019.

[4] Sergey Bartunov, Adam Santoro, Blake Richards, Luke Marris, Geoffrey E Hinton, and Timothy Lillicrap. Assessing the scalability of biologically-motivated deep learning algorithms and architectures. In *Advances in Neural Information Processing Systems*, pages 9368–9378, 2018.

[5] Sarah-Marie Belcastro and Thomas C Hull. Modelling the folding of paper into three dimensions using affine transformations. *Linear Algebra and its applications*, 348(1):273–282, 2002.

[6] Marshall Bern and Barry Hayes. The complexity of flat origami. In *SODA*, volume 96, pages 175–183. Atlanta, GA, 1996.

[7] Katia Bertoldi, Vincenzo Vitelli, Johan Christensen, and Martin van Hecke. Flexible mechanical metamaterials. *Nature Reviews Materials*, 2(11):17066, 2017.

[8] Davide Bigoni. *Nonlinear solid mechanics: bifurcation theory and material instability*. Cambridge University Press, 2012.

[9] Loïc Blanc, Alain Delchambre, and Pierre Lambert. Flexible medical devices: review of controllable stiffness solutions. In *Actuators*, volume 6, page 23. Multidisciplinary Digital Publishing Institute, 2017.

[10] Vincent Blondel and John N Tsitsiklis. Np-hardness of some linear control design problems. *SIAM Journal on Control and Optimization*, 35(6):2118–2127, 1997.

[11] Avrim L Blum and Ronald L Rivest. Training a 3-node neural network is np-complete. *Neural Networks*, 5(1):117–127, 1992.

[12] A Boubakri, Nader Haddar, K Elleuch, and Yves Bienvenu. Impact of aging conditions on mechanical properties of thermoplastic polyurethane. *Materials & Design*, 31(9):4194–4201, 2010.

[13] William L Chapman, Jerzy Rozenblit, and A Terry Bahill. System design is an np-complete problem. *Systems Engineering*, 4(3):222–229, 2001.

[14] Bryan Gin-ge Chen and Christian D Santangelo. Branches of triangulated origami near the unfolded state. *Physical Review X*, 8(1):011034, 2018.

[15] Bryan Gin-ge Chen, Nitin Upadhyaya, and Vincenzo Vitelli. Nonlinear conduction via solitons in a topological mechanical insulator. *Proc. Natl. Acad. Sci. U. S. A.*, 111(36):13004–13009, 2014.

[16] Yan Chen and Woon Huei Chai. Bifurcation of a special line and plane symmetric bricard linkage. *Mech. Mach. Theory*, 46(4):515–533, 2011.

[17] Corentin Coulais, Alberico Sabbadini, Fré Vink, and Martin van Hecke. Multi-step self-guided pathways for shape-changing metamaterials. *Nature*, 561(7724):512–515, 2018.

[18] Corentin Coulais, Eial Teomy, Koen De Reus, Yair Shokef, and Martin Van Hecke. Combinatorial design of textured mechanical metamaterials. *Nature*, 535(7613):529–532, 2016.

[19] William B Cross, Anthony H Kariotis, and Frederick J Stimler. Nitinol characterization study. *NASA*, CR-1433, 1969.

[20] Erik D Demaine and Joseph O'Rourke. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*. Cambridge University Press, 2008.

[21] YW Deng, TL Yu, and CH Ho. Effect of aging under strain on the physical properties of polyester–urethane elastomer. *Polymer journal*, 26(12):1368, 1994.

[22] Kenneth Diest. *Numerical methods for metamaterial design*, volume 127. Springer, 2013.

[23] Sam Dillavou, Menachem Stern, Andrea J Liu, and Douglas J Durian. Demonstration of decentralized, physics-driven learning. *arXiv preprint arXiv:2108.00275*, 2021.

[24] Levi H Dudte, Etienne Vouga, Tomohiro Tachi, and Lakshminarayanan Mahadevan. Programming curvature using origami tessellations. *Nature materials*, 15(5):583–588, 2016.

[25] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

[26] Bastiaan Florijn, Corentin Coulais, and Martin van Hecke. Programmable mechanical metamaterials. *Phys. Rev. Lett.*, 113(17):175503, 2014.

[27] Mukesh V Gandhi and BD Thompson. *Smart materials and structures*. Springer Science & Business Media, 1992.

[28] A Gillman, K Fuchi, and PR Buskohl. Truss-based nonlinear mechanical analysis for origami structures exhibiting bifurcation and limit point instabilities. *International Journal of Solids and Structures*, 147:80–93, 2018.

[29] Benjamin M Goldsberry and Michael R Haberman. Negative stiffness honeycombs as tunable elastic metamaterials. *Journal of Applied Physics*, 123(9):091711, 2018.

[30] Stephen Grossberg. Adaptive pattern classification and universal recoding: Ii. feedback, expectation, olfaction, illusions. *Biological cybernetics*, 23(4):187–202, 1976.

[31] Grace X Gu, Chun-Teh Chen, and Markus J Buehler. De novo composite design based on machine learning algorithm. *Extreme Mechanics Letters*, 18:19–28, 2018.

[32] Shan Zi Gui and Yukuo Nanzai. Aging in quenched poly (methyl methacrylate) under inelastic tensile strain. *Polymer journal*, 33(5):444, 2001.

[33] Paul Z Hanakata, Ekin D Cubuk, David K Campbell, and Harold S Park. Accelerated search and design of stretchable graphene kirigami using machine learning. *Physical review letters*, 121(25):255304, 2018.

[34] Richard Scheunemann Hartenberg and Jacques Denavit. *Kinematic synthesis of linkages*. McGraw-Hill, 1964.

[35] Donald Olding Hebb. *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.

[36] Markus Henke and Gerald Gerlach. On a high-potential variable-stiffness device. *Microsystem technologies*, 20(4-5):599–606, 2014.

[37] Daniel Hexner, Andrea J Liu, and Sidney R Nagel. Periodic training of creeping solids. *Proceedings of the National Academy of Sciences*, 117(50):31690–31695, 2020.

[38] Daniel Hexner, Nidhi Pashine, Andrea J Liu, and Sidney R Nagel. Effect of aging on the non-linear elasticity and memory formation in materials. *arXiv preprint arXiv:1909.00481*, 2019.

[39] Seong-Gu Hong, Keum-Oh Lee, and Soon-Bok Lee. Dynamic strain aging effect on the fatigue resistance of type 316l stainless steel. *International Journal of Fatigue*, 27(10-12):1420–1424, 2005.

[40] Larry L Howell. *Compliant mechanisms*. John Wiley & Sons, 2001.

[41] Thomas C Hull et al. Modelling the folding of paper into three dimensions using affine transformations. *Linear Algebra and its applications*, 348(1-3):273–282, 2002.

[42] Nicholas E Jackson, Michael A Webb, and Juan J de Pablo. Recent advances in machine learning towards multiscale soft materials design. *Current Opinion in Chemical Engineering*, 23:106–114, 2019.

[43] Anil K Jain, Robert P. W. Duin, and Jianchang Mao. Statistical pattern recognition: A review. *IEEE Transactions on pattern analysis and machine intelligence*, 22(1):4–37, 2000.

[44] Jing Jin, Shuangjun Chen, and Jun Zhang. Uv aging behaviour of ethylene-vinyl acetate copolymers (eva) with different vinyl acetate contents. *Polymer degradation and stability*, 95(5):725–732, 2010.

[45] Denis Jordan and Marcel Steiner. Configuration spaces of mechanical linkages. *Discrete & Computational Geometry*, 22(2):297–315, 1999.

[46] Ji-Hwan Kang, Hyunki Kim, Christian D Santangelo, and Ryan C Hayward. Enabling robust self-folding origami by pre-biasing vertex buckling direction. *Advanced Materials*, 31(39):0193006, 2019.

[47] Nathan C Keim, Joseph D Paulsen, Zorana Zeravcic, Srikanth Sastry, and Sidney R Nagel. Memory formation in matter. *Reviews of Modern Physics*, 91(3):035002, 2019.

[48] J. Kieffer. Differential analysis of bifurcations and isolated singularities for robots and mechanisms. *IEEE Transactions on Robotics and Automation*, 10(1):1–10, 1994.

[49] Jon Kieffer. Differential analysis of bifurcations and isolated singularities for robots and mechanisms. *IEEE T. Robotic. Autom.*, 10(1):1–10, 1994.

[50] Jason Z Kim, Zhixin Lu, and Danielle S Bassett. Design of large sequential conformational change in mechanical networks. *arXiv preprint arXiv:1906.08400*, 2019.

[51] Jason Z Kim, Zhixin Lu, Steven H Strogatz, and Danielle S Bassett. Conformational control of mechanical networks. *Nature Physics*, page 1, 2019.

[52] SG Konarski, MF Hamilton, and MR Haberman. Elastic nonlinearities and wave distortion in heterogeneous materials containing constrained negative stiffness inclusions. In *2014 8th International Congress on Advanced Electromagnetic Materials in Microwaves and Optics*, pages 130–132. IEEE, 2014.

[53] Sridhar Kota and GK Ananthasuresh. Designing compliant mechanisms. *Mech. Eng. CIME*, 117(11):93–97, 1995.

[54] Izabela K Kuder, Andres F Arrieta, Wolfram E Raither, and Paolo Ermanni. Variable stiffness material and structural concepts for morphing applications. *Progress in Aerospace Sciences*, 63:33–55, 2013.

[55] P Kumar and S Pellegrino. Computation of kinematic paths and bifurcation points. *International Journal of Solids and Structures*, 37(46-47):7003–7027, 2000.

[56] Harry G Kwatny, Bor-Chin Chang, and Shiu-Ping Wang. Static bifurcation in mechanical control systems. In *Bifurcation Control*, pages 67–81. Springer, 2003.

[57] Dimitris C Lagoudas. *Shape memory alloys: modeling and engineering applications*. Springer, 2008.

[58] Robert J Lang. The science of origami. *Physics world*, 20(2):30, 2007.

[59] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[60] Dong-Hyun Lee, Saizheng Zhang, Asja Fischer, and Yoshua Bengio. Difference target propagation. In *Joint european conference on machine learning and knowledge discovery in databases*, pages 498–515. Springer, 2015.

[61] ME Lee-Trimble, Ji-Hwan Kang, Ryan C Hayward, and Christian D Santangelo. Robust folding of elastic origami. *arXiv preprint arXiv:2109.10989*, 2021.

[62] Lin Li, David H Myszka, Andrew P Murray, and Charles W Wampler. Using the singularity trace to understand linkage motion characteristics. In *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V06AT07A063–V06AT07A063. Design Engineering Division, August 2013.

[63] Yongmin Liu and Xiang Zhang. Metamaterials: a new frontier of science and technology. *Chemical Society Reviews*, 40(5):2494–2507, 2011.

[64] AV Lyulin, B Vorselaars, MA Mazo, NK Balabaev, and MAJ Michels. Strain softening and hardening of amorphous polymers: Atomistic simulation of bulk mechanics and local dynamics. *EPL (Europhysics Letters)*, 71(4):618, 2005.

[65] Wei Ma, Feng Cheng, and Yongmin Liu. Deep-learning-enabled on-demand design of chiral metamaterials. *ACS nano*, 12(6):6326–6334, 2018.

[66] Patrick T Mather, Xiaofan Luo, and Ingrid A Rousseau. Shape memory polymer research. *Annual Review of Materials Research*, 39:445–471, 2009.

[67] J Michael McCarthy. *Geometric design of linkages*, volume 11. Springer Science & Business Media, 2006.

[68] Geoff McKnight and Chris Henry. Variable stiffness materials for reconfigurable surface applications. In *Smart Structures and Materials 2005: Active Materials: Behavior and Mechanics*, volume 5761, pages 119–126. International Society for Optics and Photonics, 2005.

[69] Pankaj Mehta, Marin Bukov, Ching-Hao Wang, Alexandre GR Day, Clint Richardson, Charles K Fisher, and David J Schwab. A high-bias, low-variance introduction to machine learning for physicists. *Physics Reports*, 2019.

[70] Marc Z Miskin, Kyle J Dorsey, Baris Bircan, Yimo Han, David A Muller, Paul L McEuen, and Itai Cohen. Graphene-based bimorphs for micron-sized, autonomous origami machines. *Proceedings of the National Academy of Sciences*, 115(3):466–470, 2018.

[71] Koryo Miura. Method of packaging and deployment of large membranes in space. *The Institute of Space and Astronautical Science report*, (618):1–9, 1985.

[72] Javier R Movellan. Contrastive hebbian learning in the continuous hopfield model. In *Connectionist models*, pages 10–17. Elsevier, 1991.

[73] Leonard Mullins. Softening of rubber by deformation. *Rubber chemistry and technology*, 42(1):339–362, 1969.

[74] David H Myszka, Andrew P Murray, and Charles W Wampler. Mechanism branches, turning curves, and critical points. *ASME 2012 International Design Engineering*

Technical Conferences and Computers and Information in Engineering Conference, pages 1513–1525, January 2012.

[75] David H. Myszka, Andrew P. Murray, and Charles W. Wampler. Mechanism Branches, Turning Curves, and Critical Points. Volume 4: 36th Mechanisms and Robotics Conference, Parts A and B:1513–1525, 08 2012.

[76] David H Myszka, Andrew P Murray, and Charles W Wampler. Computing the branches, singularity trace, and critical points of single Degree-of-Freedom, Closed-Loop linkages. *J. Mech. Robot.*, 6(1):011006, February 2014.

[77] Lisa M Nash, Dustin Kleckner, Alismari Read, Vincenzo Vitelli, Ari M Turner, and William TM Irvine. Topological mechanics of gyroscopic metamaterials. *Proceedings of the National Academy of Sciences*, 112(47):14495–14500, 2015.

[78] Don Norman. *The design of everyday things: Revised and expanded edition*. Basic books, 2013.

[79] Jamie K Paik, An Byoungkwon, Daniela Rus, and Robert J Wood. Robotic origamis: Self-morphing modular robot. In *ICMC*, 2012.

[80] Nidhi Pashine. Local rules for fabricating allosteric networks. *Physical Review Materials*, 5(6):065607, 2021.

[81] Nidhi Pashine, Daniel Hexner, Andrea J Liu, and Sidney R Nagel. Directed aging, memory, and nature's greed. *Science Advances*, 5(12):eaax4215, 2019.

[82] S Pellegrino. *Deployable Structures*. Springer, 2014.

[83] Edwin A Peraza-Hernandez, Darren J Hartl, Richard J Malak Jr, and Dimitris C Lagoudas. Origami-inspired active structures: a synthesis and review. *Smart Materials and Structures*, 23(9):094001, 2014.

[84] Edwin A Peraza-Hernandez, Darren J Hartl, Richard J Malak Jr, and Dimitris C Lagoudas. Origami-inspired active structures: a synthesis and review. *Smart Materials and Structures*, 23(9):094001, 2014.

[85] Michael Philen, Ying Shan, Kon-Well Wang, Charles Bakis, and Christopher Rahn. Fluidic flexible matrix composites for the tailoring of variable stiffness adaptive structures. In *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 1703, 2007.

[86] Matthew B Pinson, Menachem Stern, Alexandra Carruthers Ferrero, Thomas A Witten, Elizabeth Chen, and Arvind Murugan. Self-folding origami at any energy scale. *Nat. Commun.*, 8:15477, 18 May 2017.

[87] Matthew B Pinson, Menachem Stern, Alexandra Carruthers Ferrero, Thomas A Witten, Elizabeth Chen, and Arvind Murugan. Self-folding origami at any energy scale. *Nat. Commun.*, 8:15477, 2017.

[88] H_E Read and GA Hegemier. Strain softening of rock, soil and concrete—a review article. *Mechanics of Materials*, 3(4):271–294, 1984.

[89] Daniel R Reid, Nidhi Pashine, Justin M Wozniak, Heinrich M Jaeger, Andrea J Liu, Sidney R Nagel, and Juan J de Pablo. Auxetic metamaterials from disordered networks. *Proceedings of the National Academy of Sciences*, 115(7):E1384–E1390, 2018.

[90] Pedro M Reis, Heinrich M Jaeger, and Martin van Hecke. Designer Matter: A perspective. *Extreme Mech. Lett.*, 5:25–29, 2015.

[91] Blake A Richards, Timothy P Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, et al. A deep learning framework for neuroscience. *Nature neuroscience*, 22(11):1761–1770, 2019.

[92] D Rocklin, Vincenzo Vitelli, and Xiaoming Mao. Folding mechanisms at finite temperature. *Preprint at http://arXiv.org/abs/1802.02704*, 2018.

[93] D. Zeb Rocklin, Vincenzo Vitelli, and Xiaoming Mao. Folding mechanisms at finite temperature, 2018.

[94] Jason W Rocks, Andrea J Liu, and Eleni Katifori. The topological basis of function in flow networks. *arXiv preprint arXiv:1901.00822*, 2019.

[95] Jason W Rocks, Nidhi Pashine, Irmgard Bischofberger, Carl P Goodrich, Andrea J Liu, and Sidney R Nagel. Designing allostery-inspired response in mechanical networks. *Proceedings of the National Academy of Sciences*, 114(10):2520–2525, 2017.

[96] Jason W Rocks, Henrik Ronellenfitsch, Andrea J Liu, Sidney R Nagel, and Eleni Katifori. Limits of multifunctionality in tunable networks. *Proceedings of the National Academy of Sciences*, 116(7):2506–2511, 2019.

[97] Henrik Ronellenfitsch and Eleni Katifori. Global optimization, local adaptation, and the role of growth in distribution networks. *Physical review letters*, 117(13):138301, 2016.

[98] Henrik Ronellenfitsch, Johannes Liesche, Kaare H Jensen, N Michele Holbrook, Alexander Schulz, and Eleni Katifori. Scaling of phloem structure and optimality of photoassimilate transport in conifer needles. *Proceedings of the Royal Society B: Biological Sciences*, 282(1801):20141863, 2015.

[99] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[100] Christian D Santangelo. Extreme mechanics: Self-folding origami. *Annu. Rev. Conden. Ma. P.*, 8:165–183, 2017.

[101] Adrien Saremi and Zeb Rocklin. Topological elasticity of flexible structures. *Physical Review X*, 10(1):011052, 2020.

[102] Benjamin Scellier and Yoshua Bengio. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24, 2017.

[103] Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24):241722, 2018.

[104] Jesse L Silverberg, Arthur A Evans, Lauren McLeod, Ryan C Hayward, Thomas Hull, Christian D Santangelo, and Itai Cohen. Using origami design principles to fold reprogrammable mechanical metamaterials. *Science*, 345(6197):647–650, 2014.

[105] 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*, 117(26):14843–14850, 2020.

[106] Menachem Stern, Sam Dillavou, Marc Z. Miskin, Douglas J. Durian, and Andrea J. Liu. Physical learning beyond the quasistatic limit. *Phys. Rev. Research*, 4:L022037, May 2022.

[107] 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*, 11(2):021045, 2021.

[108] Menachem Stern, Viraaj Jayaram, and Arvind Murugan. Shaping the topology of folding pathways in mechanical systems. *Nature communications*, 9(1):1–8, 2018.

[109] Menachem Stern, Matthew B Pinson, and Arvind Murugan. The complexity of folding self-folding origami. *Phys. Rev. X*, 7(4):041070, 2017.

[110] Menachem Stern, Matthew B Pinson, and Arvind Murugan. Continual learning of multiple memories in mechanical networks. *Physical Review X*, 10(3):031044, 2020.

[111] Tomohiro Tachi. Generalization of rigid-foldable quadrilateral-mesh origami. *Journal of the International Association for Shell and Spatial Structures*, 50(3):173–179, 2009.

[112] Tomohiro Tachi. Geometric considerations for the design of rigid origami structures. In *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium*, volume 12, pages 458–460. Elsevier Ltd, 2010.

[113] Tomohiro Tachi. Geometric considerations for the design of rigid origami structures. In *Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium*, volume 12, pages 458–460, 2010.

[114] Tomohiro Tachi. One-dof cylindrical deployable structures with rigid quadrilateral panels. *Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures: Proceedings*, 2010.

[115] Tomohiro Tachi and Thomas C Hull. Self-foldability of rigid origami. *Journal of Mechanisms and Robotics*, 9(2):021008, 2017.

[116] Tsvi Tlusty, Albert Libchaber, and Jean-Pierre Eckmann. Physical model of the genotype-to-phenotype map of proteins. *Physical Review X*, 7(2):021037, 2017.

[117] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[118] Scott Waitukaitis, Rémi Menaut, Bryan Gin-ge Chen, and Martin Van Hecke. Origami multistability: From single vertices to metasheets. *Physical review letters*, 114(5):055503, 2015.

[119] Charles W Wampler. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. *IEEE T. Syst. Man Cyb.*, 16(1):93–101, 1986.

[120] Charles W Wampler, Jonathan D Hauenstein, and Andrew J Sommese. Mechanism mobility and a local dimension test. *Mechanism and machine theory*, 46(9):1193–1206, 2011.

[121] Kjell Westra, Francis Dunne, Stasia Kulsa, Mathew Hunt, and Jacob Leachman. Compliant polymer origami bellows in cryogenics. *Cryogenics*, 114:103226, 2021.

[122] Logan G Wright, Tatsuhiro Onodera, Martin M Stein, Tianyu Wang, Darren T Schachter, Zoey Hu, and Peter L McMahon. Deep physical neural networks trained with backpropagation. *Nature*, 601(7894):549–555, 2022.

[123] JF Wycoff, S Dillavou, M Stern, AJ Liu, and DJ Durian. Desynchronous learning in a physics-driven learning network. *The Journal of Chemical Physics*, 156(14):144903, 2022.

[124] Le Yan, Riccardo Ravasio, Carolina Brito, and Matthieu Wyart. Architecture and coevolution of allosteric materials. *Proceedings of the National Academy of Sciences*, 114(10):2526–2531, 2017.

[125] Ying Zhou, Carlos M Duque, Christian D Santangelo, and Ryan C Hayward. Biasing buckling direction in shape-programmable hydrogel sheets with through-thickness gradients. *Advanced Functional Materials*, 29(48):1905273, 2019.