Symmetries in String Theory

Application of Machine Learning to Theoretical Physics

Marc Syväri

SIGILL. UNIVERSITAT. LUDOVIC. MAXIM.

München 2021

Symmetries in String Theory

Application of Machine Learning to Theoretical Physics

Marc Syväri

Dissertation
der Fakultät für Physik
der Ludwig-Maximilians-Universität
München

vorgelegt von
Marc Syväri
aus Berlin

München, den 13. August 2021

Erstgutachter: Prof. Dr. Dieter Lüst
Zweitgutachter: PD Dr. Michael Haack
Tag der mündlichen Prüfung: 28. September 2021

Zusammenfassung

In dieser Doktorarbeit werden wir verschiedene neue Methoden entwickeln, um Symmetrien, mit Hilfe von maschinellem Lernen, zu erkennen und auf Probleme der theoretischen Physik, insbesondere der Stringtheorie, anzuwenden.

Diese Arbeit ist in zwei Teile unterteilt. Im ersten Teil konzentrieren wir uns auf die Entwicklung von, auf maschinelles Lernen gestützte, Werkzeuge zur Symmetrie-Erkennung, während wir uns im zweiten darauf fokussieren, die Effekte bereits bekannter Symmetrien der Stringtheorie zu untersuchen.

Im ersten Abschnitt dieser Arbeit geben wir eine kurze Einführung in die wichtigsten Bereiche und Methoden des maschinellen Lernens. Im Anschluss an diesen Review stellen wir die verschiedenen von uns entwickelten Tools vor. Die erste Idee verwendet Strukturen in den verborgenen Schichten des neuronalen Netzwerkes, die bei dem Trainieren des Neuronalen Netzwerkes entstehen. Datenpunkte, die durch eine Symmetrie verbunden sind, werden in den verborgenen Schichten gruppiert und können dadurch benutzt werden, um zusammengehörige Punkte zu finden. Der zweite Ansatz ist die direkte Suche von Erhaltungsgrößen eines Systems durch eine Koordinatentransformation zu „teilweise“ zyklischen Koordinaten und damit die direkte Bestimmung von Erhaltungsgrößen. Die dritte Idee ist die Konstruktion von Lax-Paaren für verschiedene Systeme. Durch die Definition einer geeigneten Verlustfunktion können wir ein neuronales Netz konstruieren, das analytische Formeln für Lax-Paare bestimmen kann. Lax-Paare sind die definierende Eigenschaft integrierbarer Systeme. Diese Operatoren-Paare ermöglichen es uns, Erhaltungsgrößen zu berechnen und Informationen über die Stabilität eines Systems zu erhalten.

Im zweiten Teil dieser Arbeit untersuchen wir die Effekte von Dualitäten in der Stringtheorie. Wir geben einen Überblick über die Symmetrie-Gruppen der S-Dualität und der T-Dualität, sowie ihre Vereinigung, der U-Dualität. Im nächsten Schritt betrachten wir die exzeptionelle Feldtheorie, die manifest invariant unter der U-Dualität ist. Durch diese finden wir neue lokal nicht-geometrische Räume der M-Theorie, die bisher unbekannte lokal nicht-geometrischer R-Flüsse enthalten. Im Folgenden konstruieren wir Dualitätsketten, die Informationen über jene neuen Räume enthalten, die fehlende Impulse in Verbindung mit den neuen nicht-geometrischen R-Flüssen aufweisen. Im letzten Kapitel untersuchen wir den Phasenraum offener String-Endpunkte. Für einen lokal nicht-geometrische String-Hintergrund zeigen wir, dass der Phasenraum nicht-assoziativ ist.

Abstract

In this thesis, we will develop new machine learning based methods to detect symmetries and use these tools on different problem settings in theoretical physics, specifically in string theory.

This work is divided into two main parts. One is focused on the development of machine learning tools for symmetries, whereas in the second part we focus on examining the effects of already known symmetries of string theory.

In the first part, we give a light introduction to the most important areas and methods of machine learning. Following this review, we present the different tools we developed. The first idea uses structures in the hidden layers of a neural network. Data points linked by symmetries are clustered in the training process of the neural network and therefore can be used to find points connected by a symmetry when examining the representation of the data points in the hidden layers. The second approach is the direct search for conserved quantities of a system through a coordinate transformation to "partially" cyclic coordinates and thus, the direct search of conserved quantities. The third mechanism constructs Lax pairs for various systems. By defining a suitable loss function, we can construct a neural network which gives us an analytical formula for a Lax pair. Lax pairs are the defining property of integrable systems. These pairs of operators allow us to compute conserved quantities and provide information on the stability of a system.

In the second part, we investigate the effects of dualities in string theory. We give an overview of the symmetry groups of the S-duality and the T-duality, as well as their unification, the U-duality. In the next step we introduce exceptional field theory, which is manifestly invariant under the U-duality. Through this theory we find new locally non-geometric spaces of M-theory, which contain previously unknown locally non-geometric R-fluxes. In the following we construct duality chains that provide information about those new spaces that have missing momentum modes in connection with the new non-geometric R-fluxes. In the last chapter we examine the phase space of open string endpoints. For locally non-geometric string backgrounds we show that the phase space is non-associative.

Acknowledgments

First overall, I want to thank my doctoral supervisor Dieter Lüst for giving me the opportunity to work in his research group for nearly five years (including my master thesis). It was a pleasure to work in his group and to have the chance to work in two different areas of string theory. Whenever I had questions regarding research or related to academic life, he supported me and gave helpful advice.

I also want to express my gratitude to Emanuel Malek and Sven Krippendorf for the daily discussions and support during my time in the string theory group. Emanuel for introducing me to the academic world and helping me with my first steps doing research in string theory. Sven for working the last nearly two and a half years closely together with hours of discussions on research ideas and projects, but also just for getting a coffee together and talking about life outside university. I want to thank him for the time and effort to share his knowledge on machine learning with me and the chance to ask any question. It was a great time exploring the opportunities of machine learning in string theory together.

Additionally, I want to thank all the people who made my time at the chair unforgettable. Let me thank my fellow PhD students who created a great atmosphere and made my time in the group unforgettable. A special thanks to two groups of people: David, Fabrizio and Nicolas for sharing the office for a too short period of time. You created the best office atmosphere possible. And to David, Felix and Rene for keeping company during the pandemic. The visits at the coffee machine enlightened my day during these weird times. Equally, I want to thank the administration of the Max Planck Institute, especially Monika Goldammer and Annette Sturm for helping me whenever necessary.

Apart from the members of the group I want to thank Max, Emi and Thomas for discussing physics and making (not only) academic life much more enjoyable.

I also want to thank Adrian, David, Emi, Fabrizio, Max, Ralf and my father Jyrki for valuable comments and proof-reading parts of this thesis.

In the final paragraph I want to thank all my friends for good company, support and sharing beautiful moments together during my time as a student. In particular, I am thankful for the support of my girlfriend Sandra during my PhD and the stress connected to it (especially while writing this thesis). Last but not least, my deepest gratitude to my family for the constant support and encouragement during my life and my time as a PhD student.

Contents

Part I

Introduction

Chapter 1

Motivation

Before outlining the main ideas of this thesis, let us motivate the central idea of the thesis: Using machine learning techniques to discover symmetries within datasets. We give an overview on the recent success of machine learning showing above human performance in many different fields. Then, we motivate the focus on symmetries due to the large impact these had in the development of modern high energy physics, and conclude with an overview on string theory, the leading candidate for a unification of gravity and particle physics. We will conclude with the outline of this thesis.

1.1 The Success of Machine Learning

The idea to create a computer with a kind of intelligence comparable to humans is an old dream of computer scientists. For this reason, Alan Turing created the Turing test in 1948 to classify what artificial intelligence is [1]: A computer has to be able to convince a human that it is a human and not a computer. The first definition of a neuron was already given in 1943 [2], and lead to the development of the first perceptrons [3]. In combination with the first back-propagation algorithms [4], which were applied to multi-layer perceptrons in 1980 [5], they lead to the development of different tools for image recognition and natural language processing. One of the biggest break-throughs was the development of convolutional neural network by AT&T Bell Laboratories for digit recognition [6]: Applied at the US Postal service, it was the beginning of the wide spread use of deep neural networks. After this success, the interest from computer scientists increased with the availability of computing power and in 2006, the Netflix prize was launched to improve the recommendation algorithm from Netflix [7]. The release of the database "ImageNet" in 2009 led to an increasing publicity and further accelerated research [8]. In 2016, it was the first time a computer program – the program AlphaGo – was able to beat an unhandicapped human player in the game of Go [9]. Together with the development of the package TensorFlow, machine learning approaches became common. They improved existing solutions in the fields like natural language processing, face recognition as well as image and video generation, and showed above human-level performances in all of these fields. After the success in computer science, machine learning provided the tools for many problems in natural sciences, specifically in physics. A good overview over the recent progress in the different fields of physics can be found in [10].

In this thesis, we focus on the development of methods to detect and identify symmetries because they play a crucial role not only in string theory but all over physics.

1.2 Symmetries as General Guideline in Physics

Solving problems in physics is often connected to finding a symmetry of the system and therefore, using redundancies of the problem. A typical example is the use of spherical coordinates which simplifies the notion of many systems.

The idea of using symmetries to improve the understanding of nature drives the era of modern physics from the beginning. Galileo Galilei can be seen as the founder of this approach. He realized first that physical laws should be the same, independent of the reference frame, as long as they differ only by a constant relative motion. Transformations between such two reference frames, together with spatial rotations and translations, are known as Galilean transformation. The insights in moving reference frames led to a first notion of a basic principle of relativity and allowed him to understand the physical laws of nature and resulting for example in the development of the laws of the free fall. This principle of no absolute motion or absolute rest provided the framework for Newton to find "Newton's laws of motion". Additionally, his idea of equivalence of the gravitational forces on earth and between planets and the sun can be viewed as an extension of the idea that laws should be generalizable over different physical systems that are related by symmetries.

The next important step towards a modern physical understanding of the world was the unification of electrodynamics and magnetism. They are described by Maxwell's equations and show the symmetry between the two different forces (up to sources of the fields). This symmetry led to the claim of the existence of magnetic monopoles. Those monopoles would explain for example the quantization of charges as shown by Dirac [11] and are a good example of the power of symmetries to conjecture new phenomena, even though they have not been discovered as real particles.

An extension to Galilei's basic principle of relativity led to the next big step in the development of modern physics: At the end of the 19th century, it was known that light has a finite speed. Albert Einstein combined this knowledge with conjecturing the basic principle of relativity, namely that the speed of light should be the same, independent of the reference frame [12]. As a consequence, time and space were no longer independent from each other, but merged into a single entity called space-time described by the Lorentz symmetry group. This resulted in the theory of special relativity.

Again, we notice a close connection between unification and symmetry group. A unification is often physically motivated. To describe them properly and understand the consequences of it, it is necessary to find the underlying symmetry of a unified system.

The next idea of unification was to include gravity to the basic principle of relativity. Inspired by the Eötvös experiment [13], which measured the correlation between the inertial mass and the gravitational mass, Einstein included the gravitational force to the basic principle of relativity and developed the “Gedankenexperiment” of a free-falling elevator in a gravitational field. Assuming the equivalence of vacuum and the free fall in a gravitational field, nowadays called (strong) equivalence principle, this “Gedankenexperiment” led to the development of general relativity [14].

The theory is the key building block to explain many phenomena of gravity in the universe, and an important starting point to build new theories.

The second building block of modern physics is quantum mechanics. Here, every force is mediated by particle interactions. Max Planck noticed that energy of the radiation emitted by a black body takes discrete values [15]. Albert Einstein understood that these can be understood as particles [16]. The development in the field of quantum mechanics and the understanding of the interaction between photons and electrons eventually resulted in the first example of a so called quantum field theory – quantum electrodynamics – with the symmetry group $U(1)_Q$. In these theories classical fields and particles are combined into a single fundamental object, namely a quantum field. In the following decades the theoretical background of gauge theories and the other fundamental forces was developed. It turned out that all of them can be described by quantum field theories where each is based upon a certain symmetry group. For the electroweak force, a combination of electromagnetism and the weak force, the underlying symmetry is $U(1)_Y \times \mathrm{SU}(2)_L$, for the strong force it is the symmetry group $\mathrm{SU}(3)_c$. As a result, we find the symmetry group of the $U(1) \times SU(2) \times SU(3)$ for the Standard Model of particle physics. After the discovery of the Higgs boson in 2012, all particles of the Standard Model are now found [17].

Parallel to the search for the remaining particles, physicists tried to further unify the electroweak and strong interaction while also focusing on a top-down approach to unify gravity with the other forces.

For the unification, a sufficiently large symmetry group has to be chosen which then gets spontaneously broken to the symmetry group of the Standard Model. These theories are called Grand Unified Theories (GUT). The simplest symmetry group for a GUT is SU(5). It is the smallest group which contains the symmetry group of the Standard Model. Unfortunately, these theories predict an instability of the proton which requires the scale of symmetry breaking to be very high, i.e. at 10^{15} - 10^{16} GeV, way out of the observable range of today’s colliders [18].

The top-down approach, namely string theory, is described in more detail in the next section. In string theory, the fundamental element are strings, one-dimensional objects which exist in ten dimensions and therefore the additional dimensions have to be “removed” which leads to many different low energy theories.

Overall, we hope this short review of the history of modern particle physics shows the importance of symmetries in theories in the development of modern physics. It is not only important in the development of modern physics, but also the key building block to examine new theories. Therefore, it is a necessity to develop the new tools to find such symmetries easier and faster.

1.3 String Theory

Let us give an introduction into String theory (we follow the textbook material of [19]): Originally an idea to explain quantum-chromodynamics, it soon turned out that it contains not only additional massless particles leading to other forces within the theory but also spin-2 particles which lead to a graviton within the theory.

Therefore, a good candidate was found which connected the two main building blocks of high energy physics: general relativity and the Standard Model of particle physics.

The general idea is quite straight forward: we expand the zero-dimensional point particles to one-dimensional strings. In contrast to the one-dimensional world line of a point particle (the trajectory within the spacetime), strings have a two-dimensional worldsheet described by the Nambu-Goto action:

$$\begin{aligned} S_{\text{NG}} &= -T \int_{\Sigma} \mathrm{d}A\,, \\ &= -T \int_{\Sigma} \mathrm{d}^2\sigma \left(-\det{}_{\alpha\beta} \left(\frac{\partial X^{\mu}}{\partial \sigma^{\alpha}} \frac{\partial X^{\nu}}{\partial \sigma^{\beta}} \eta_{\mu\nu} \right) \right)^{1/2} , \end{aligned} \tag{1.1}$$

with Σ being the worldsheet and $\sigma^{\alpha} = (\sigma, \tau)$ are the coordinates parameterizing the worldsheet. The fields $X^{\mu} = X^{\mu}(\sigma, \tau)$ embeds the worldsheet into the flat target space with metric $\eta_{\mu\nu}$. $T = \frac{1}{2\pi\alpha'}$ is the string tension, a constant of mass dimension two. α' is the so-called Regge slope and connects the string tension with the string length scale and the string mass scale:

$$\ell_s = 2\pi\sqrt{\alpha'} \qquad \text{and} \qquad M_s = \left(\alpha'\right)^{-1/2} . \tag{1.2}$$

To remove the square-root in the action and quantize it, we introduce the Polyakov action in conformal gauge:

$$\begin{aligned} S_P &= -\frac{T}{2} \int d^2\sigma\, \eta^{\alpha\beta} \partial_{\alpha} X^{\mu} \partial_{\beta} X_{\mu} \\ &= 2\,T \int d\sigma^{+} d\sigma^{-}\, \partial_{+} X \cdot \partial_{-} X\,. \end{aligned} \tag{1.3}$$

In the second line, we use the light cone gauge with left and right moving coordinates $\sigma^{\pm} = \tau \pm \sigma$.

The Polyakov action and the corresponding string solutions can be canonical quantized. An important consequence of the quantization is that the bosonic string has a critical dimension of $D = 26$ for the target space making it necessary to find a mechanism to reduce the number of dimensions. With this number of dimensions, the proposed graviton becomes massless and is therefore a valid candidate. Additionally, the bosonic string theory contains scalar tachyons, objects with negative mass-squared which lead to an unstable theory.

The next step towards a consistent theory is to construct a string containing bosonic but also fermionic fields. This leads to the superstring action. Quantizing the superstring leads

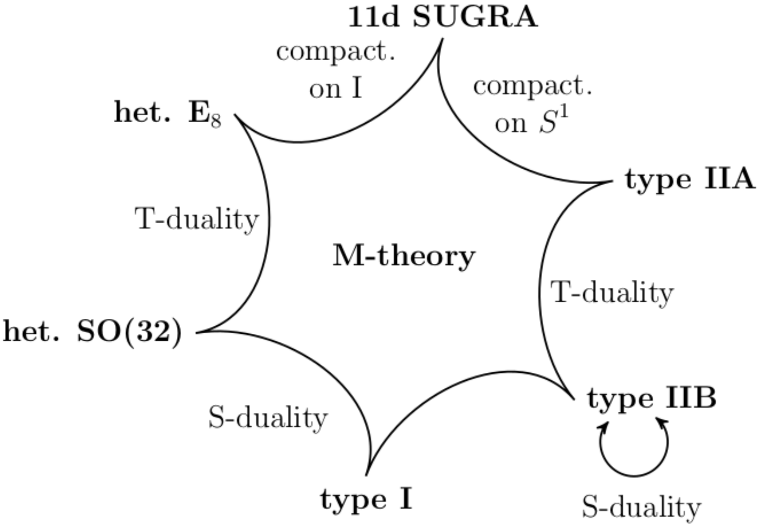


Figure 1.1: Connection between the different string theories by different dualities. S-duality is an inversion of the string coupling g_s, T-duality is an inversion of the radius of one of the compactified directions, and compact. on $I/$ or S^1 is the compactification over an interval or respectively a circle (in the section 7.3 we explain the details of these dualities). Taken from [21].

to an interesting effect on the spectrum of the string: We get two different sectors of the string for left movers and right movers, namely the NS-sector and the R-sector. It can be shown that a string theory containing all combinations is inconsistent. Therefore, we have to truncate the spectrum to find a consistent supersymmetric theory. The GSO (Gliozzi-Scherk-Olive) projection [20] ensures that while demanding supersymmetry the tachyonic states get truncated, and we end up with two consistent, supersymmetric tachyon-free string theories: the **Type IIA** and **Type IIB** string theory. For those theories the critical dimension is $D = 10$.

Additionally to the two Type II string theories, we can construct other string theories in ten dimensions.

- **Type I** string theory describes an unoriented string in 10 dimensions. It not only contains closed strings but also necessarily open strings. It is an $\mathcal{N} = 1$ supergravity which couples to SO(32) supersymmetric Yang-Mills theory.

- **Heterotic** string theory couples a superstring in $D = 10$ dimensions as right mover to a bosonic string with $D = 26$ dimensions as left mover. To couple those, we have to compactify 16 dimensions of the bosonic string to match the number of dimensions. For the compactification two symmetry groups are possible, the SO(16) symmetry group as well as the $E_8 \times E_8$-group resulting in two additional symmetry groups.

- **Type 0A** and **Type 0B** string theory describes two non-supersymmetric and unstable theories. Therefore, they are not consistent and therefore, cannot used to construct a suitable theory containing the Standard Model of particle physics and general relativity.

In 1995 Witten suggested in a talk that these string theories are all connected to each other by dualities between the theories. Including the eleven-dimensional supergravity theory, which is a classical description of a supersymmetric theory in the maximal allowed number of spacetime dimensions, we find a web of theories which are connected to each other. The underlying eleven-dimensional theory is called M-theory. Unfortunately, there does not exist a complete non-perturbative formulation of it. Therefore, it is object of current research to better understand the structure and the phase space. Figure 1.1 shows the different theories and how they are connected to each other. A detailed description of the different dualities can be found in section 7.3. Note that from the perspective of M-theory those dualities would be symmetries within the theory. In part III, we examine the effects of those symmetries on the phase space.

Chapter 2

Outline and Summary

In this thesis, we focus on both sides of symmetries: In part II, we focus on the identification of symmetries in different physical systems using machine learning to develop different frameworks. In part III, we examine the effect of symmetries and dualities in string theory. We are able to identify consequences for the phase space only detectable using duality chains to examine non-geometric spaces. We conclude in part IV with an outlook.

The results are based on the author's publications [22, 23, 24, 25, 26], the individual chapters are organized as follows:

- We start with an review of the basic ideas and methods from machine learning. At first, we start with a general overview of the different concepts within machine learning and continue with the most important ingredients which are necessary to define neural networks. Here, we focus on the parts of machine learning which are necessary to understand the other chapters of this thesis.

- In the third chapter, we present a method which is able to identify symmetries and other similar structures within a dataset. When training a neural network on a classification task, the hidden layers of the neural network encode structure information of the dataset. This knowledge is used to identify the presence of a symmetry and find invariant points under those symmetries. We present applications on different potentials, as well as a class of Calabi-Yau manifolds. In a second step, we develop an algorithm to explicitly identify the symmetry group.

- Next, we construct a neural network which automatically finds conserved quantities from trajectories of particles. As a prerequisite we describe how to find the Hamiltonian, then we build a neural network which provides coordinate transformation to cyclic coordinates. In these coordinate systems, the generalized momenta are conserved. Identifying conserved quantities allows us to understand the symmetries of such systems by Noether's theorem.

- After identifying conserved quantities in classical mechanics, chapter 6 is dedicated to integrability of field theories and, as a direct effect, their conserved quantities. In field theories, we focus on computing Lax Pairs, which are necessary to compute the infinite tower of conserved quantities of such theories. We are able to distinguish between integrable and non-integrable perturbations of integrable systems and provide a framework to scan over different types of perturbations. In general, the framework provides a method for the search of highly constrained systems of differential equations.

- In the third part, we start in chapter 7 with a review of the different types of dualities between the different string theories and their mathematical formulation as symmetry groups. Following the idea of making theories invariant under symmetry transformation we review the construction of exceptional field theory, an extended theory which is invariant under S- and T-dualities. We describe the construction of the transformation laws of U-duality, the analogue to the Buscher rules of T-duality. We conclude with the known duality chains for non-geometric backgrounds in double field theory and in four-dimensional exceptional field theory.

- Chapter 8 is based on the results of [25]. We focus on the construction of high-dimensional duality chains and find previously unknown non-geometric fluxes using exceptional field theory. Lying in the R-R sector of string theory these non-geometric fluxes go beyond the previously known R-fluxes from the NS-NS sector. Constructing those non-geometric backgrounds from twisted tori, we find missing momenta modes in the phase space associated to the non-geometric R-fluxes.

- In the last chapter of part 3, we focus on the phase space of the open string. Previously, it was only known for closed strings that their phase space is non-associative in non-geometric spaces. Starting from a D0-brane on a three-dimensional torus with H-flux, we compute the commutation relations for the end points along the T-duality chain. For non-geometric R-flux backgrounds, we find a non-associative phase space for the end points as well.

We also want to point out the author's recent publication [27]. Although not part of this thesis, it yields additional methods how to examine the structure of the string vacua landscape and can therefore provide additional methods to search for suitable string vacua.

Part II

Methods to Identify Symmetries

Chapter 3

Introduction to Machine Learning

In this section, we give an overview over the different techniques of machine learning and the components necessary when training a neural network. For an extensive review, we refer to the text books [28, 29] for a general introduction and to the reviews [30, 31] for an introduction for physicists.

This section is organized as follows: First, we give an overview over the different segments of machine learning. Then we describe the different parts of a neural network as well as optimizers and loss functions, and end up with the example of convolutional neural networks in section 3.5.

3.1 Basics of Machine Learning

Machine learning can be broadly categorized into three concepts: supervised learning, unsupervised learning and reinforcement learning. They differ in the underlying ideas and it depends on the type of problem setting which one to use and how the model has to be trained. A characterization of each of them is given below, later, we are going to focus on unsupervised learning to find unknown structure in data.

- **Supervised Learning:** The model is trained on examples where the output is known. The goal of the training process is to learn a map which finds the correct output for the different inputs and should be as general as possible. The performance is measured on an unseen test set. We distinguish between regression, where the target is a number (or a vector), and classification learning. Here, a typical example is image classification, for example MNIST (see figure 3.1 for more information).

- **Unsupervised Learning:** The model is trained purely on input data, and no output labels are given leaving the neural network the task to structure the data. Typical tasks are discovering patterns in data, but also finding meaningful features in feature learning.

- **Reinforcement Learning:** The model interacts with an environment and performs different actions while trying to achieve a goal (e.g. playing a video game, driving a car etc.) The model navigates through a (usually) high dimensional phase space, while it gets rewards from the environment. It is a mixture of supervised and unsupervised learning: the goal is known, but it is unclear how to achieve it. A good introduction to this method is [32].[1]

In this section, we give an example how to train a neural network using supervised learning in a linear regression task – to detect symmetries, we are mainly using unsupervised learning. In most physical problem settings the symmetries are a-priori unknown and have to be found by the neural network itself. Therefore unsupervised learning is the best choice, but compared to supervised learning the main building blocks remain the same. The only exception is chapter 4: Here, we use a side effect of the supervised learning in a neural network to find symmetries in a neural network.

Let's start with the rough idea of the training procedure in the case of supervised learning, where we will get in touch with all important components which will be explained in detail in section 3.2.

- We start with the dataset: Every data point (typically denoted by x) we want to classify comes with a label $\hat{y}$. The label $\hat{y}$ is the target value. A typical example is MNIST, where x are images of hand-written digits, the label $\hat{y}$ is then the number represented on the pictures. Such a data set usually has between 10^4 and 10^5 pairs of input data and labels. Due to the fact that this is too large to use them in a network all at once, we divide the data sets into shuffled batches of a fixed size and use the batches instead of all inputs for the training. One period is a training run with all batches. After each period, the batches are re-shuffled. Typically, we split the data set into two parts: the training set, on which the neural network is trained, and the test set, which the network does not use for training, but only to measure the performance of the neural network.

- These batches are fed into the neural network. We will later explain how these networks are constructed, and which ingredients play a role. For the moment, it is important that they are functions $f_\theta(x)$, which map the input x to the output space. The computation of $f_\theta(x)$ is called the forward pass because the information is going forward through the neural network. These functions $f_\theta(x)$ depend on a high number of (depending on the problem 10^3 - 10^9) parameters denoted by θ. The goal is to optimize these parameters such that the neural network is able to predict the right class labels as output.

- After the forward pass, we calculate the error between the predicted class and the true value (i.e. the label). A typical loss function for regression tasks is the Mean Squared Error (MSE):

$$\mathcal{L}_{\text{MSE}} = 1\frac{1}{n_{\text{batch}}} \sum_{i\in\text{batch}} \|f_\theta(x_i) - \hat{y}_i\|_2^2\,, \tag{3.1}$$

---

[1] It is typically used for games or robots, but the author along with others showed in [33] that it can also be used in the context of string theory to navigate through the string vacua landscape.

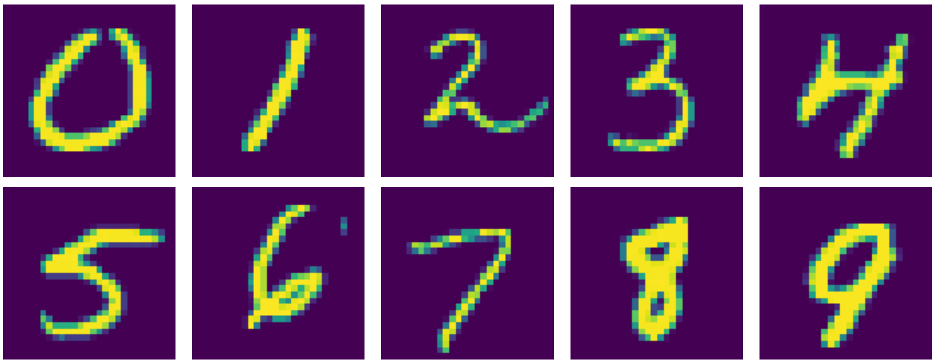


Figure 3.1: Examples of the MNIST dataset [34]. It contains 70,000 images of hand-written digits and was one of the first benchmark datasets for image recognition.

where n_{batches} denotes the number of samples per batch. Other typical choices are the mean absolute error for regression tasks and binary/categorical cross-entropy for classification tasks.

- After computing the prediction error of the neural network, we have to optimize the weights of the neural network to improve the results. The simplest ansatz for the optimizer is gradient descent: we compute the derivative of the loss with respect to the weights itself and change their values in this direction:

$$\theta_i' = \theta_i - \eta \nabla_{\theta_i} \mathcal{L} \,, \tag{3.2}$$

  where η is the learning rate which defines the step size of the gradient descent. As a hyper parameter the value of η has to be fixed before training. There are many improvements for the optimizer such as using momentum or an adaptive learning rate. The whole process is called back-propagation due to the fact that the information is now going backwards through the neural network. The last three steps describe one complete training step. For one episode, we have to repeat these three steps for every batch of the data set.

- After the training process is completed (or even between two episodes) we have to test the performance of the neural network. Therefore, we compute the output of the neural network on the test data set and compares it to the true labels. The difference gives you the performance of the neural network.

This general framework can be also used for unsupervised learning. Here, one has to use an abstract definition of the loss function. Let us demonstrate this procedure with a simple example.

Note that in general two different types of problems exist when fitting a neural network. One is called underfitting. In this case, the neural network is not able to find the right function because the number of weights is too small or because of other reasons (wrong activation function, wrong layers or a badly designed loss function). In this case, the training loss is a good guidance, and it should go to zero in the training process (or at least become very small). A typical example from statistics is trying to fit a third order polynomial with only a second order polynomial. We won't be able to find a suitable fit because we used the wrong function. The opposite problem is overfitting. When fitting a polynomial with little noise and N data points, we are able to find a degree $N-1$ polynomial which passes every data point, but the generalization would be quite bad because of the oscillation between the data points. This problem exists for neural networks as well: when the neural network has too many free weights, it might be able to fit the training dataset with perfect accuracy, but shows a bad performance on the test dataset. For this thesis, we deal with this problem in two different ways: In chapter 4, we have to solve this problem of finding the right number of generators. Here, we use a principle component analysis to find the right number of generators. In the chapters 5 and 6, we are using a synthetic dataset with an arbitrarily large amount of data to avoid this problem. Additionally, we start there with a small number of free parameters and take more parameters into account up to the point where the error converges to zero. This slow raising in the number of degrees of freedom is possible because we are actually looking for analytical solutions, which is not expected to be a high order polynomial.

3.1.1 Linear regression

We apply the presented procedure to a simple example of linear regression. The goal is to fit the function

$$f(x) = 2\,x + 1\,. \tag{3.3}$$

As data points we use all integers $0 \leq x \leq 10$. Therefore, we have the dataset with the points:

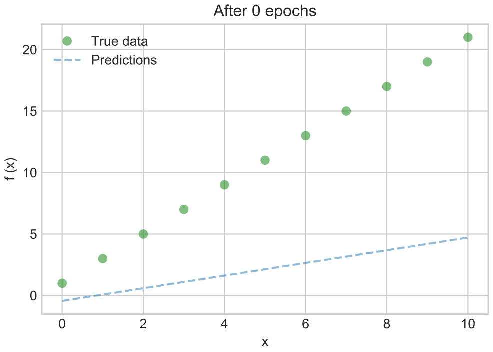
x	0	1	2	3	4	5	6	7	8	9	10
$\hat{y}$	1	3	5	7	9	11	13	15	17	19	21

Because the dataset is quite small, we use the whole dataset as one batch. As toy neural network we take a neural network with two parameters $\theta = (w, b)$ and perform a linear regression on the equation

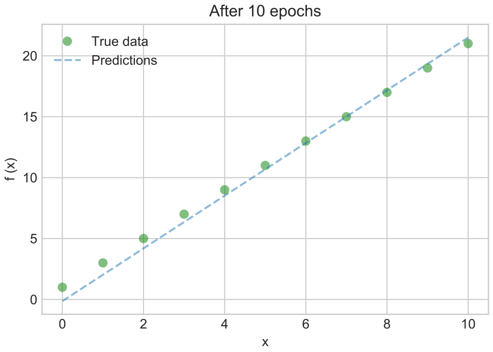
$$f_\theta\left(x_{\text{input}}\right) = w\; x_{\text{input}} + b\,. \tag{3.4}$$

The weights are chosen from a normal distributions $\mathcal{N}\left(0, 1\right)$. As loss function we take the mean-squared error (3.1), and use gradient descent (3.2) to update the weights with $\eta = 0.01$.

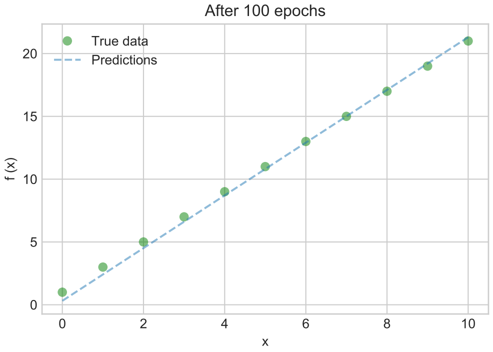
We can now start the training. Note that every training looks different because we draw the weights from a normal distribution. The function f_θ before the training is shown in the picture below as a solid line, the points represent our data points:



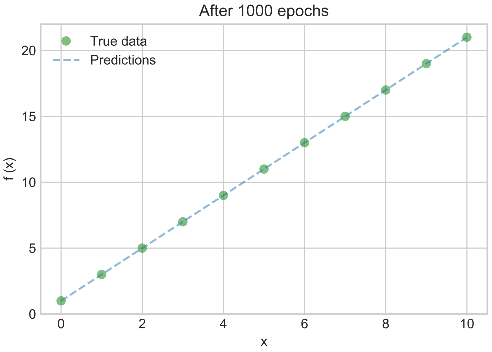
After 10 steps, we can see a clear improvement of the function f_θ:



After 100 steps, the function is pretty accurate:



To get a higher precision of the numbers, we have to increase the number of epochs. With gradient descent we get a nearly perfect result after 1000 epochs:



The formula we found is

$$f_\theta \left(x_{\text{input}} \right) = 2.0006 \; x_{\text{input}} + 0.9956 \,. \tag{3.5}$$

In this simple example, the run time per episode is short, but for larger networks and data it becomes unfeasible to use such a high number of training epochs. We have to improve the efficiency of the algorithms by choosing appropriate layer structure, adjusted learning rates and activation functions (see the next chapters). Gradient descent becomes inefficient at points close to a local minimum of the function because here the gradient becomes really small. Later in section 3.4, we present more sophisticated optimizers to avoid this problem. After this short example to present the basic concept, we move on to the different components used nowadays in machine learning.

3.2 Network Architecture

As previously mentioned, neural networks are highly-parameterized functions and the challenge is how to construct those functions. It turned out that the human brain is a good role model to abstract certain properties and construct neural networks from those. In brains, synapses and neurons are the smallest building blocks. They get information from previous neurons, combine it linearly, and then perform a non-linear operation (called activation function or non-linearity) on the value of the function. Finally, the information is passed on to the next row of neurons.

Feed-forward neural networks are the most common type, where the information is only passed in one direction in the neural network. Feed-forward neural networks consist of a set of layers, where each layer has multiple neurons and they only take information from the previous layer and pass it to the next layer. Therefore, no information between neurons can be exchanged within one layers. We distinguish between the input layer, the hidden layer and the output layer (see Figure 3.2 for an example of a neural network with labeled layers).

- The input layer is the first layer of the neural network. Here, no computation is performed - it is used to pass on the information to the hidden layers.

- Hidden layers are all layers inside the neural network. The values of the nodes are not visible from outside. In these layers he neural network extracts the essential information and passes them from the input layer to the output layer.
- The output layer is the final layer of the neural network. It depicts the results of the computation from the neural network and passes it to the user.

The simplest layer is the linear layer. It is also known as fully-connected layer or dense-layer. As the name indicates in this layer any neuron is connected with all neurons in the previous one. All the weights of the connections are independent, which allows us to write the value of the μ-th neuron of the i-th layer in the following way:

$$z_{\mu}^{(i)} = \sum_{\nu=1}^{n_{i-1}} w_{\mu\nu}^{(i)} y_{\nu}^{(i-1)} + b_{\mu}^{(i)} \tag{3.6}$$

Here, $w_{\mu\nu}^{(i)}$ and $b_{\mu}^{(i)}$ are trainable weights of the neural network, and n_i is the number of neurons in the i-th layer. Afterwards, we have to apply the activation function σ_i to each neuron:

$$y_{\mu}^{(i)} = \sigma_i \left(z_{\mu}^{(i)} \right) \qquad \text{for all } i \text{ and } \mu\,. \tag{3.7}$$

Note that the activation function might be different for every layer, but we do not use different activation functions within the same layer as this leads to different scales in the back-propagation. Additionally, we define the "zero-th" layer as the input and the final layer as the output of the network function f_θ. Therefore, the defining equations for a neural network with s layers are:

$$\begin{aligned} y^{(0)} &= x_{\text{input}}\,, \\ z_{\mu}^{(i)} &= \sigma_i \left(\sum_{\nu=1}^{n_{i-1}} w_{\mu\nu}^{(i)} y_{\nu}^{(i-1)} + b_{\mu}^{(i)} \right)\,, \\ f_\theta\left(x_{\text{input}}\right) &= y^{(s)}\,. \end{aligned} \tag{3.8}$$

In Figure 3.2 we can see a simple example of such a feed-forward neural network. Combining equations (3.8), we get the complete output function of the neural network with parameters $w_{\mu\nu}^{(i)}$ and $b_{\mu}^{(i)}$. We might think of this network function as some interesting basis for the function with complicated weight constraints. In this thesis all of the neural networks are constructed using linear layers, but there exist a vast range of different neural networks, such as Convolutional layers and Pooling layers (described in 3.5) for image classification and long short-term memory layers (LSTM-layer) and transformer layers for natural language processing. We will continue defining the different activation function.

3.2.1 Activation Functions

Typically, a layer consists of linear transformation. If we would only use such linear transformations, the whole network could be represented by a single linear transformation and

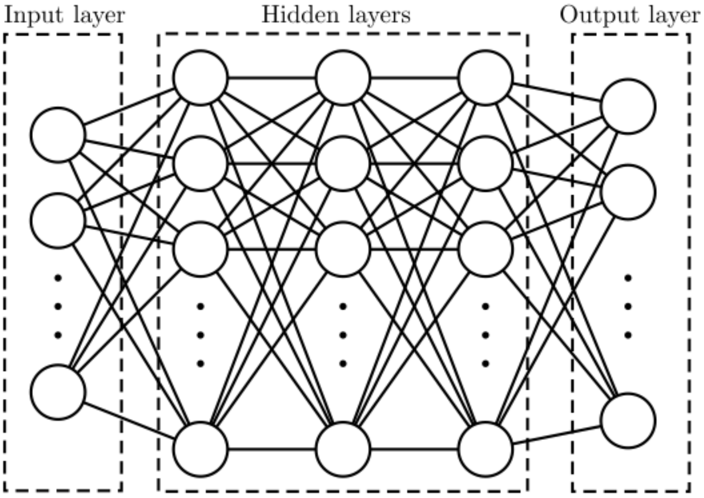


Figure 3.2: A simple schematic Multi-layer Perceptron (MLP) with the different layers. Here, we have one input, one output and three hidden layers. Image from [31]

no hidden layer would be necessary. Therefore, it is necessary to add non-linear functions. There exists a rich variety of non-linearities. It can be shown that all non-polynomial functions are sufficient for approximations within a neural network [35]. Typically, one uses unary function (functions with only one input value) which acts on each neuron individually to speed up the computations within a neural network (except for the softmax-activation). Typical choices are:

- ReLU: $\sigma : \mathbb{R} \mapsto \mathbb{R}^+ ,$ $\sigma(x) = \max(0, x)$
- Leaky ReLU: $\sigma : \mathbb{R} \mapsto \mathbb{R} ,$ $\sigma(x) = \max(cx, x)$ with $c > 0$
- Logistic sigmoid: $\sigma : \mathbb{R} \mapsto (0,1) ,$ $\sigma(x) = \frac{1}{1+e^{-x}}$
- Tanh: $\sigma : \mathbb{R} \mapsto (-1,1) ,$ $\sigma(x) = \tanh(x)$
- softmax: $\sigma : \mathbb{R}^n \mapsto (0,1) ,$ $\sigma(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$

The softmax function is used to normalize the output with respect to the other neurons. Due to this normalization the neurons sum up to one, which is used for classification. The value of the neurons gives the probability for the individual class. We give now a more detailed descriptions for the different activation functions.

- **(Leaky) ReLU:** ReLU stands for Rectified Linear Unit. Nowadays, it is the most popular activation function due to its easy computation of the function itself, as well as the derivative which speeds up the calculations. One disadvantage of the ReLU activation is that for negative values no information is passed at all and therefore, important information might get lost. This can be solved by using the Leaky ReLU-activation: It is the same for positive numbers, but instead of being 0 for all negative numbers it has a small constant slope (usually smaller then 0.01). Note that the

ReLU-activation is only non-zero for first order derivatives. Higher derivatives, often required in physics, always vanish.

- **Logistic sigmoid:** One of the first activation functions and the typical representative of the sigmoid activation functions: It maps the input in the interval between 0 and 1 and is monotonically increasing. The advantage of the logistic sigmoid is that the derivative can be represented using the logistic sigmoid function as well:

  $$\frac{d}{dx}\sigma\left(x\right) = \sigma\left(x\right)\left(1 - \sigma\left(x\right)\right) \, . \tag{3.9}$$

  It is not only used as an activation function, but also often used for binary classification due to the fact that it maps to values between 0 and 1 which can be interpreted as probabilities.

- **Tanh:** The tanh-activation function is a shifted and rescaled sigmoid function, and therefore, has similar properties as the logistic sigmoid function.

- **Softmax:** The softmax-function is usually not used within a neural network. It is mainly used to normalize the values of the output layers in such a way that they are summed up to 1 and can be interpreted as probabilities. Therefore, it is commonly used in neural networks for classification and can be viewed as a “multi-dimensional sigmoid function”.

3.2.2 Multi-Layer perceptron

One of the basic neural networks is the Perceptron [3]. In its most simple form it has no hidden layers and takes an n-dimensional input x_i. Then the output value o_j is defined by:

$$o_j = \begin{cases} 1, & \sum_i w_{ij} x_i + b > 0 \, , \\ 0, & \text{else} \, . \end{cases} \tag{3.10}$$

Here, the activation function is defined using a threshold b. Such a perceptron is actually able to learn the functions AND, OR and NOT, but for example it cannot represent the function XOR. Replacing the threshold by a sigmoid function, and using 0.5 as threshold, we can represent the one-layer perceptron using a combination of a linear layer and the activation function sigmoid. Extending this idea to multiple hidden linear layers of arbitrary size with activation functions, we call it a **multi-layer perceptron** which is represented by the function (3.8). Such multi-layer perceptrons are now able to learn functions like XOR. A mathematically interesting property is that already a multi-layer perceptron with one hidden layer and an infinite number of neurons can fit any function [35], which makes neural networks universal approximators. This theoretical result can be extended to an network with arbitrary depth [36]. These results give a theoretical understanding of the power of neural networks. We are going to use multi-layer perceptrons in most of our applications (with a finite number of hidden layers and neurons). Even though they are one of the simplest neural networks, multi-layer perceptrons have the advantage of not assuming any bias for the the input data, unlike convolutional neural networks for example do (see section 3.5).

3.3 Loss Functions

Another key element is the loss function: This function defines the punishment of the neural network in the training process. It is usually defined in such a way that for a perfectly correct answer the loss function is zero. Note, that negative values as reward are also possible, but not common in supervised learning in contrast to deep reinforcement learning. Therefore, in most cases we could simply use a mean-squared error (MSE), but it turns out that often there exist much more efficient choices. We will present the most common choices for loss functions in supervised learning. In the formulae we are going to use $\hat{y}_i$ as target value and $y_i = f(x_i)$ as the output of the neural network. The index i labels the sample in the batch, n_{batch} refers to the batch size.

- **Mean-Squared Error (MSE)**: The most simple choice which can be used in nearly every situation. Typically, it is used in regression tasks. The loss is given by

  $$L_{MSE}\left(y, \hat{y}\right) = \frac{1}{n_{\text{batch}}} \sum_{i=1}^{n_{\text{batch}}} \left(y_i - \hat{y}_i\right)^2 . \tag{3.11}$$

  It has the advantage that it tries to minimize outliers and is therefore a good loss function for a clean data set where it is known that there are no mistakes within the data set.

- Another loss is the **Mean-Absolute Error (MAE)**. It is computed by

  $$L_{MAE}\left(y, \hat{y}\right) = \frac{1}{n_{\text{batch}}} \sum_{i=1}^{n_{\text{batch}}} \left|y_i - \hat{y}_i\right| . \tag{3.12}$$

  It has the advantage that it is more robust to outliers and tries to minimize the error of as many points as possible rather than minimizing outliers. One of the disadvantages is that the norm of the gradient does not correspond to the distance to the minimum, but it always has the same size. Therefore, using MAE as a loss function often leads to an oscillating behavior around the minimum. One alternative is the Huber loss. It uses for small values of the loss a quadratic loss term, whereas it is linear for higher values.

- Cross entropy losses are typically used for classification problems: Both the target value and the output value are seen as probability distribution. Therefore, the target is usually encoded as a one-hot vector (zeros everywhere except for a single one which labels the correct class). The output probabilities describe how certain the neural network is regarding a specific class label. In general, the **cross entropy** between two probability distributions P and Q can be computed as

$$S(P,Q) = \mathbb{E}_{X\sim P}\left[-\log_2 Q(x)\right] , \tag{3.13}$$

where $x \sim P$ denotes that x is drawn from probability distribution P. In the discrete case, this reduces to

$$S(P,Q) = \sum_i p_i \ln\left(q_i\right) , \tag{3.14}$$

where p_i and q_i are the probabilities for the discrete class i. In the case of a binary classification where $y \in \{0,1\}$ (a yes or no answer) we can use **binary cross entropy** (BCE) as loss function

$$L_{\mathrm{BCE}}\left(y,\hat{y}\right) = -\left(y\ln\left(\hat{y}\right) + (1-y)\ln\left(1-\hat{y}\right)\right) . \tag{3.15}$$

Note that one of the terms always vanishes for a specific choice of y. This notion can be extended for multi-class classification to the **categorical cross-entropy**, given by

$$L_{\mathrm{CCE}}\left(y,\hat{y}\right) = -\sum_a y_{c_a,x}\ln\hat{y}_{c_a,x} , \tag{3.16}$$

where $y_{c_a,x}$ denotes the probability value c_a for a given input x, whereas $\hat{y}_{c_a,x}$ is the network output (usually normalized using a softmax layer). Also for the categorical cross-entropy all terms except for one vanish due to the one-hot encoding.

Modifying loss functions allows us to restrict the output of the neural network and to define abstract properties within the set of solutions. Therefore, we are going to define specific loss functions in section 5 and 6 to find symmetries.

3.4 Optimizer

After introducing the most important layers and defining typical loss functions for neural networks, let us introduce the last part of the training algorithm: The optimizer. In the beginning the weights of the neural network are randomly initialized. To learn a good approximation function $f_\theta\left(x_{input}\right)$, the weights have to be updated in the right way. Optimizers provide a mathematical algorithm how to update the weights and find a solution to the given problem in a finite amount of time. In total, we can think of the problem as a high-dimensional landscape in the parameter space where we try to find the minimum in a fast and reliable manner. Here, we will describe the basic and most used ways to navigate through this landscape.

- The most straight forward method is the **gradient descent**. The idea is to go always in the steepest direction downhill with finite step size, so going along ∇L where L is the total loss. The free parameter is the learning rate η which is the proportional constant of the step size. Mathematically, the change for parameter θ_i with learning rate η is

$$\begin{aligned} \Delta\omega_t &= \nabla_{\theta_t} L\left(\theta_t, x(i), y(i)\right) \,, \\ \theta_{t+1} &= \theta_t - \eta \; \Delta\omega_t \,. \end{aligned} \tag{3.17}$$

The loss L depends on all parameters at step t as well as input values $x(i)$ and target values $y(i)$. Challenging problems are the existence of local minima and saddle points where the algorithm might get stuck. One solution for these problems is the **stochastic gradient descent** (SGD): Due to the usage of randomized batches of samples, where the composition of samples changes per episode, the derivative is always computed for different samples. Therefore the exact position for local minima and saddle points changes. This avoids getting stuck over multiple time steps.

- Another problem are slopes with a small gradient. The SGD might need too long to traverse through such a slope. The solution is to add a momentum term to the SGD. This allows the optimizer to get some speed when multiple steps are going in the same direction. It also helps not getting stuck in local minima or at saddle points, even though it does not avoid this problem in every case. The mathematical notation for the **stochastic gradient descent with momentum** is

$$\begin{aligned} \Delta\omega_t &= \beta \; \Delta\omega_{t-1} + \nabla_{\theta_t} L\left(\theta_t, x(i), y(i)\right) \\ \theta_{t+1} &= \theta_t - \eta \; \Delta\omega_t \,. \end{aligned} \tag{3.18}$$

  η is again the learning rate, while β controls the influence of the the momentum. Typical choices are $\beta \approx 0.9$ and $\Delta\omega_0 = 0$.

- Another option is to adapt the learning rate for the individual parameters depending on the properties of the parameters. For flat directions, we want to use a large learning rate whereas for steep direction a smaller one is better. A way of enableing this is to use higher momenta and to normalize the learning rate accordingly. There exist multiple optimizers using this method. The most well known are AdaGrad [37], AdaDelta [38] and RMS-Prob [39]. The mathematical notation for RMS-Prob is:

$$\begin{aligned} g_t &= \nabla_{\theta_t} L\left(\theta_t, x(i), y(i)\right) \,, \\ s_t &= \beta \; s_{t-1} + (1-\beta) g_t^2 \,, \\ \theta_{t+1} &= \theta_t - \eta \; \frac{g_t}{s_t + \epsilon} \qquad \text{with } \epsilon \approx 10^{-8} \,. \end{aligned} \tag{3.19}$$

  β is usually around 0.9 and controls the averaging time of the second momentum whereas ϵ prevents divergence.

- Nowadays, the most common optimizer is **Adam (Adaptive Moment Estimation)**: it combines the two approaches above and uses first and second momentum to optimize the weights [40]:

$$
\begin{aligned}
g_t &= \nabla_{\theta_t} L\left(\theta_t, x(i), y(i)\right) , \\
m_t &= \beta_1\ m_{t-1} + (1-\beta_1) g_t , \\
\hat{m}_t &= \frac{m_t}{1-(\beta_1)^t} , \\
s_t &= \beta_2\ s_{t-1} + (1-\beta_2) g_t^2 , \\
\hat{s}_t &= \frac{s_t}{1-(\beta_2)^t} , \\
\theta_{t+1} &= \theta_t - \eta\ \frac{\hat{m}_t}{\hat{s}_t + \epsilon} \qquad \text{with } \epsilon \approx 10^{-8} .
\end{aligned}
\tag{3.20}
$$

β_1 and β_2 are the control factors for the averaing over time of the gradients and the second moments of the gradients, respectively. Typical values are $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

3.5 Example: The CNN

In this thesis we are going to use the previously presented multi-layer perceptron, but we want to give an impression over another type of neural network.

Convolutional Neural Networks (CNNs) [41] are widely used for image recognition. Even though more elaborated neural networks exist nowadays, such as ResNet, CNNs are an interesting starting point to get an understanding of the development of neural networks.

When applying simple feed-forward neural networks to images with $28 \times 28 = 784$ input neurons, we run into the problem that the neural networks generalize pretty badly because even shifting the image by one pixel results in a completely different situation for the neural network. An alternative here are Convolutional Neural Networks. Here, the structure uses the spatial information of the input data (e.g. which pixels are close to each other) as well as the shift symmetry of images. Generally, it consists of two parts: a **convolutional layer**, followed by a **pooling layer**. The idea is to move a two-dimensional kernel with weights over the input, performing locally a matrix multiplication (See Figure 3.3 for a graphical example). This keeps the structure of the input after the convolution intact while we are moving to more abstract features of the picture. Using the same weights on every position of the image ensures that a feature only needs to be learned at one position and can be generalized to any other. Afterwards, we apply an activation function to each neuron.

Pooling layers on the other hand reduce the number of existing neurons. For example, a 2×2 Pooling layer takes the maximum or the average sum of each 2×2-cluster, and therefore, helps to filter out specific features from the image, while reducing the computational cost.

After several layers of alternating convolutional and pooling layers, one uses linear layers combined with activation functions to find the right classes for each picture. A sample architecture can be seen in Figure 3.4.

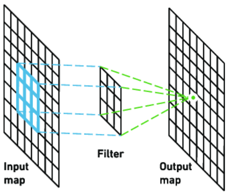


Figure 3.3: **Left:** The operation mode of a convolutional layer. The filter is moved over the input image and is trainable. There exist several ways how to pad the boundary regions. The visualisation is taken from [42]

We stop here with the review of the fundamental components of machine learning, however the interested reader is encouraged to find more explanations and examples of the vast variety of different neural network architectures in [28, 29, 30, 31]. We will conclude with an example of the training of such a convolutional neural network.

3.5.1 Training of an exemplary network

As next step, we train a sample neural network to classify MNIST to see the simplicity of the structure which is necessary to construct an accurate image classifier. The first layer is a two dimensional convolutional layer with 10 kernels, each of size 5×5 with a ReLU-activation. Then, we use a MaxPooling layer with a pooling size of 2×2. The second block is again a convolutional layer with 20 kernels with size 5×5 with a ReLU-activation, followed by a MaxPooling layer (again, with pooling size of 2×2). After this, we use two fully connected layers, the first one with 50 neurons and ReLU-activation, the second one with 10 neurons and softmax-Activation. This is the output layer used to classify the images. As loss, we used categorical cross entropy, and trained the network using stochastic gradient descent with momentum and a learning rate of 0.01. As performance on the test set we find an accuracy of 98.5 percent. The loss can be seen in this graph:

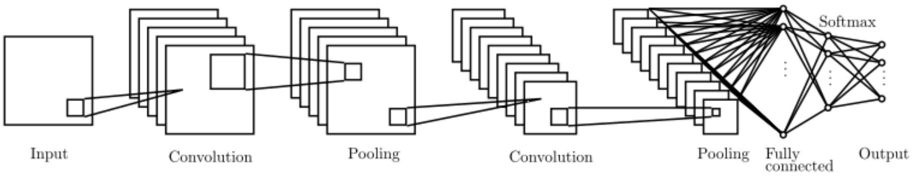
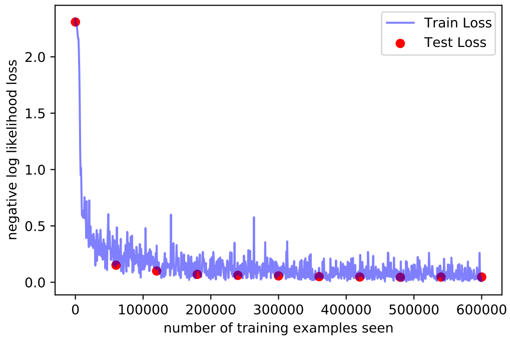


Figure 3.4: **Right:** A sample architecture for CNNs. The visualisation is taken from [31]



In each epoch the neural network is trained on 60,000 images. Therefore, after 3 epochs it already reaches an accuracy of 97.7 percent. This small example explains the success of neural network: With this simple structure and fast training neural networks solve complicated problems which would be hard to implement using classical algorithms. For our problem settings, finding the right definition of the loss function might be much harder in the beginning, but once they are defined it should be straightforward to apply them to more difficult systems.

Chapter 4

Detecting Symmetries with Neural Networks

In this chapter, we describe a general method how we can find symmetries and identify them using neural networks. In the first part, we describe how we are able to identify orbits connected by a symmetry. The idea is to train the neural network on a classification task and use the structure of the data within the embedding layers of the neural network. Data points connected by a symmetry are mapped together in the embedding space. This allows us to identify the orbits of the points under the symmetry transformation. In the second part we present an algorithm how to identify the generators using the points from the orbits found in the first section. Knowing the generators enables us to identify the symmetry group itself. We are going to present several examples on the rotation group SO(n), as well as SU(2). Additionally, we use the method presented in the first section to classify Complete Intersection Calabi-Yau manifolds. This section is based on the results of [22].
This chapter gives a first way to identify symmetrical structure within datasets. It is based on the idea that similar points are mapped together in the embedding space and that we can utilize this structure.

4.1 Finding Symmetries

As first step, we present the method how to identify previously "unknown" symmetries within a dataset by examining the clustering behavior in the embedding layer. Our technique is suitable for data questions where we have an input point x_{input} which is linked to a fixed target $f\left(x_{\text{input}}\right)$. The presence of a symmetry denotes that we have a map S which maps

$$S \,:\, x \mapsto S(x)\,, \tag{4.1}$$

which leaves the value of the function f invariant, e.g.

$$f\left(x_{\text{input}}\right) = f\left(S\left(x_{\text{input}}\right)\right)\,. \tag{4.2}$$

The key idea which we utilize to find symmetries is the property of neural networks that the embedding structures in the hidden layers often possess meaningful representations. The most well known example is described in the word2vec-paper [43], where it was shown that the neural networks learn meaningful relations between different words, so for example it was able to learn the equations

$$\begin{aligned} \vec{v}_{\text{emb}}\left(\text{“king”}\right) - \vec{v}_{\text{emb}}\left(\text{“man”}\right) &= \vec{v}_{\text{emb}}\left(\text{“queen”}\right) - \vec{v}_{\text{emb}}\left(\text{“woman”}\right)\ , \\ \vec{v}_{\text{emb}}\left(\text{“Germany”}\right) - \vec{v}_{\text{emb}}\left(\text{“Berlin”}\right) &= \vec{v}_{\text{emb}}\left(\text{“France”}\right) - \vec{v}_{\text{emb}}\left(\text{“Paris”}\right)\ , \end{aligned} \tag{4.3}$$

where $\vec{v}_{\text{emb}}\left(\text{“word”}\right)$ denotes the embedding vector of the word “word”. An anologous idea was used in [44] to find similarities between chemical elements. The authors were able to rediscover the periodic table of chemical elements by training the neural network to predict the binding energy of chemical compounds. As a result elements of the same group are clustered together.

We are going to use a similar approach to detect symmetries: We train a neural network on a supervised learning task to predict the value of the potential at a given position. Our key finding is that neural networks map points together which are linked by a symmetry. Such points are clustered together in the embedding space (i.e. the second to last layer in our neural networks). Points with the same value in the classification but not connected by a symmetry are not clustered together. This is a first step to identify the presence of a symmetry. We are going to discuss two types of symmetries here: at first, we are going to examine two problems with underlying continuous symmetries, namely the Mexican hat potential (which has an SO(2)-symmetry) and an SU(2)-invariant superpotential. Then, as an additional example, we study discrete symmetries in the context of Complete Intersection Calabi-Yau manifolds (CICYs). The procedure is in both cases the same:

- We construct a classification problem for the given problem setting.
  - For the potentials, we define classes which are defined by a respective value (plus a certain width of the class). Note, that the distance between the classes should be larger than the width of the classes.
  - For the CICYs, we use the Hodge numbers $h^{1,1}$ and $h^{1,2}$ as classification targets.

  In our experiments we find that this technique functions only in classification tasks, but not in regression tasks.
- We are going to sample points for the different classes from a uniform distribution. A balanced dataset is preferred.
- As next step, we train a neural network for the given classification task. Here, the neural network needs the capacity to train up to a reasonable accuracy (above 95 percent). In our examples, we use multi-layer perceptrons as neural networks.
- As last step, we analyze the representation of the data points the neural network is trained on. In all of the examples we find a gap between the data points connected by

a symmetry and points not connected by one. Note, that we use different strategies to find those clusters. For the potentials, we are using a dimensional reduction like t-SNE [45] to visually find the structures, whereas for the CICYs this is not possible due to the large number of different classes. Here, we analyze the structure given by the distances between the individual clusters.

4.1.1 Continuous Symmetries

Mexican Hat Potential

The first problem we look at is a two-dimensional Mexican hat potential which has an underlying SO(2)-symmetry:

$$V(x,y) = -a \cdot (x^2+y^2) + (x^2+y^2)^2 = -a \cdot r^2 + r^4\,, \tag{4.4}$$

where we use $a = 2.3$ for our numerical experiments. Here, two types of points with the same value of the potential (4.4) appear: points, which are connected by a symmetry, but also points which belong to a different cluster and are not connected to the first type of points by a symmetry. A visualization can be found on the right of Figure 4.1.

The next step is to formulate a classification problem for the potential. We define eleven classes for the value of the potential. The values of these classes are defined by

$$\left[\frac{k}{5} - 10^{-3}, \frac{k}{5} + 10^{-3}\right] \qquad k = -5, ..., 5\,. \tag{4.5}$$

Note the ratio between the width of the potential and the distance between the classes. This enables a well converging training process. The larger the width of the classes is the more complicated is the training process.

The sampling process can be implemented in a straight forward way: We randomly pick points (x, y) and check the values of the potential, whether they belong to one of the classes. We construct a balanced training set with ~ 1000 representatives per class. We use a multilayer perceptron with 7 linear layers each consisting of 80 neurons with ReLU-activation and a final layer with 11 output neurons with a softmax activation function. We train the weights using the Adam optimizer, as a loss function we use categorical crossentropy. Note that the neural network is rather oversized but this enables a fast convergence. We train the network to an accuracy above 95 percent on the training set. For our analyses we are only interested in correct classified points and therefore, we do not use a test or validation set.

Using only correct classified data points, we analyze the representation of these points in the 80-dimensional representation space. This space can be visualized using t-SNE [45] which is a method to find local structure in high dimensional spaces. Therefore, we are able to project the structure of the space in a two-dimensional plane. See Figure 4.1 for the projection of the representation.

Comparing specific classes, we find only one specific cluster for values larger than 0 (the yellow points) . Therefore, all of these points are connected by a symmetry. This is exactly

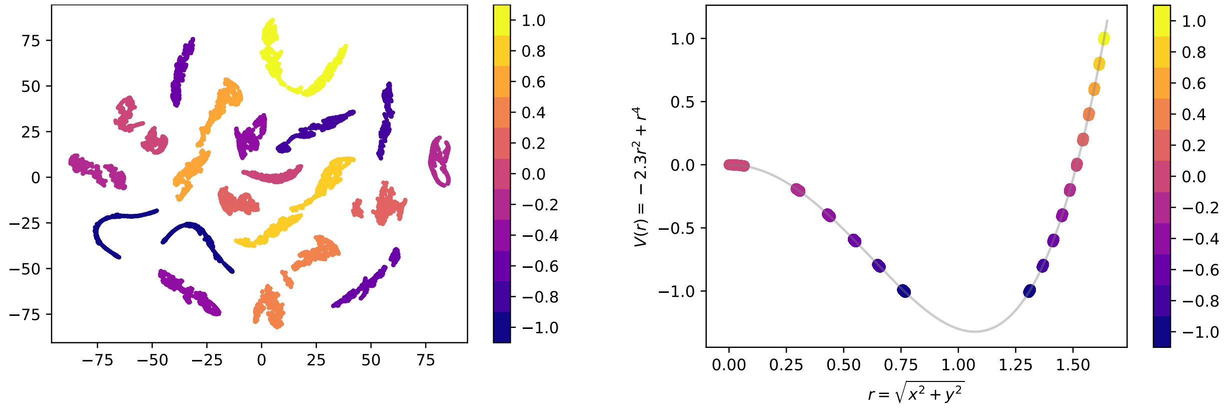


Figure 4.1: **Left:** Structure of the embedding layer using t-SNE representation (perplexity of 50). Some of these classes (indicated by the same color) have two distinct point clouds. **Right:** The Mexican hat potential with the different classes. The classes have the same color scheme as in the left picture. Using this color scheme we can match the classes with two distinct point clouds on the left to the classes on the right with two types of points.

what we expected: Looking at the right side of Figure 4.1 we only have data points of the same class for large values of the potential. For classes with values smaller or equal than zero we find a different picture: For those points we find two distinct clusters, separated by the norm of the points. This point can be made very precise: points bigger than the norm of the minimum of the potential are separated from points with a norm smaller than $r = \sqrt{a/2}$. Comparing the left and the right side of Figure 4.1, one can clearly see this split.

SU(2)-invariant Superpotential

After demonstrating this technique for the well-known Mexican hat potential, we demonstrate the method to find symmetries on an example with an SU(2)-symmetry. For this, we use the function

$$W(x,y) = (x_1 y_2 - x_2 y_1) + \frac{1}{2}(x_1 y_2 - x_2 y_1)^2\,, \tag{4.6}$$

where $x = (x_1, x_2), y = (y_1, y_2) \in \mathbb{C}^2$ and transform in the fundamental representation and the anti-fundamental representation of SU(2). Such holomorphic functions for instance appear in supersymmetric field theories and are used in many places in theoretical physics. Note, that we have two scaling symmetries in the dataset which are an additional symmetry of our system

$$\begin{aligned} x_1 &\to a\,x_1 & x_2 &\to b\,x_2 \\ y_2 &\to \frac{1}{a}\,y_2 & y_1 &\to \frac{1}{b}\,y_1\,, \end{aligned} \tag{4.7}$$

where $a, b \neq 0$.

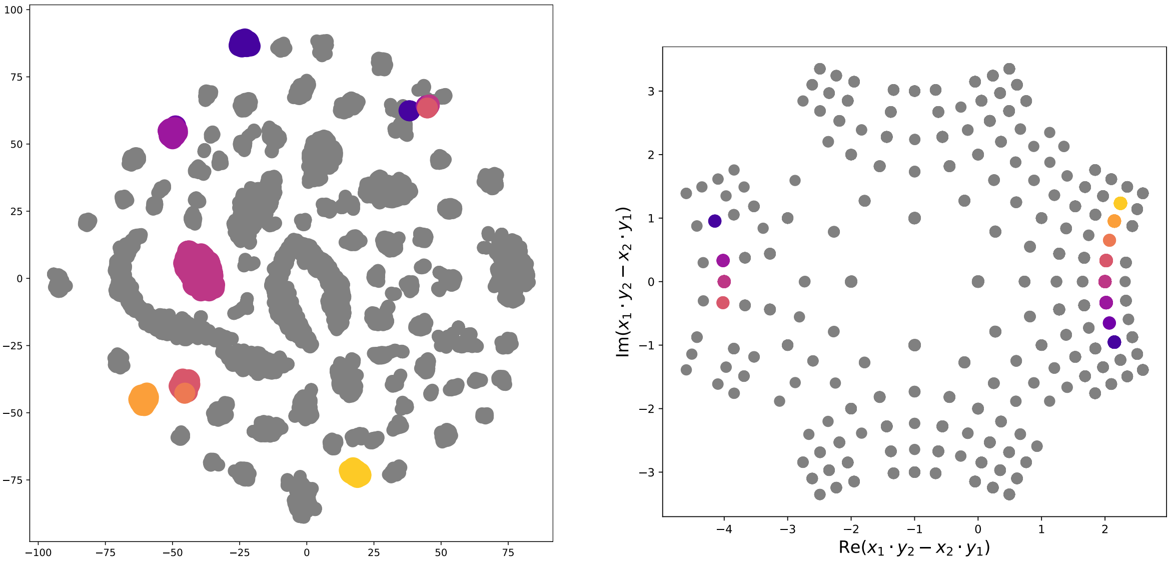


Figure 4.2: **Left:** The t-SNE representation (perplexity of 40) of the 60-dimensional embedding space. The colored clusters in the plot highlight the same classes as in the plot on the right. Gray dots mark the other points in the embedding space. Note that due to the large number of different clusters the structure is less explicit than in Figure 4.1. **Right:** Plot of the $SU(2)$ invariant quantity $x_1 \cdot y_2 - x_2 \cdot y_1$. Some of the classes have only one distinct representative (for example the yellow and the light orange class), others have two distinct classes (the dark blue/violet classes). This structure is revealed in the embedding space as we can see on the left.

The procedure is similar as for the Mexican hat potential: We start by defining classes for the values of the potential. Because the potential (4.6) is complex, whereas most neural networks are built only with real numbers we work with the real and the imaginary part independently (so basically working rather with $\mathbb{R}^2$ than with $\mathbb{C}$) and define classes for both parts. Our classes are distributed symmetrically around zero:

$$\left[k - 10^{-2}, k + 10^{-2}\right] \qquad k = -5, ..., 5\,. \tag{4.8}$$

Note, that the ratio between width and distance between the classes is of the same order as for the Mexican hat potential.

The next step is the sampling process. Again, we use a uniform distribution to sample values for x^1, x^2, y^1 and y^2, compute the values of the potential and check whether they belong to one of the classes. We sample ~ 1000 representatives per class to have a balanced data set as well. The neural network consists of 7 dense layers with ReLu activation with 60 neurons each, followed by two independent output layers with a softmax activation which are trained to find the class labels. The second to last layer (which both output layers share) is the embedding layer of the neural network. We again use the Adam optimizer combined with a categorical crossentropy on both outputs. We trained the neural network to a precision above 95 percent. To analyze the structure in the embedding space we used the t-SNE framework to reduce the 60 dimensional embedding space to a two dimensional plane which we can analyze visually. In Figure 4.2 we can see the results of the experiment: In this case we have 121 distinct classes which makes it tough to see the structure completely such as for the Mexican hat potential. Therefore, we highlighted only some of the classes. In the picture on the right we plotted the real and the imaginary part of the SU(2) invariant quantity $x_1 \cdot y_2 - x_2 \cdot y_1$ of the different points from the dataset. The colors display the value for the potential. We can see that some data points share the value of the potential, even though they have a different value of $x_1 \cdot y_2 - x_2 \cdot y_1$ (and therefore are not connected under a SU(2) transformation). On the left hand side, we see the t-SNE projection. We colored the same classes as on the right. Comparing the two pictures we can see that the structure in the embedding layers reveals the symmetry structure of the problem. Note that we performed this check for all the different classes (not only the colored ones). We are able to find the conserved quantity of the dataset and therefore of the potential.

As a consistency check we applied also the t-SNE projection on the input space to check whether such structures exist already there, but we are not able to find those in the input space.

4.1.2 Discrete Case: Identifying distinct string theory vacua

After these two continuous examples we now discuss a discrete problem arising from string theory: finding relations between string vacua and whether they are physically the same.

In this section we examine a broad class of Calabi-Yaus, so called Complete Intersection Calabi-Yaus (CICYs). These manifolds have been already classified in three dimensions [46], as well as in four dimensions [47]. Even though they have been classified it is unclear how to generalize these results to higher dimensions. It took roughly one hour to classify all three-dimensional CICYs, whereas the time needed for four dimensions was 7487 hours (roughly 312 days). Therefore, it is unfeasible to classify the five-dimensional CICYs using such classical algorithms. Here, we use the three-dimensional CICYs to test our method and show that we can extend this method from continuous potentials to much more challenging discrete problem setting. Let us start to give a short review on CICYs with the basic information we need for our analyses. Note, that we do not need a deep understanding of the exact structure, therefore it can be skipped while still understanding the purpose of this part. A much longer review can be found in [48] with more mathematical details.

Configuration matrices of CICYs

CICYs are described using configuration matrices. They describe classes of polynomials in projective spaces where the manifolds itself are the zeros of those polynomials. To give a basic introduction on them we follow [47]. An example of such a configuration matrix is

$$\left[\begin{array}{c|cc} 1 & 1 & 1 \\ 2 & 1 & 2 \\ 3 & 0 & 4 \end{array}\right] . \tag{4.9}$$

The first column denotes the dimensions of the projective spaces. Here, in this example the equations are in the product space $\mathbb{P}^1 \times \mathbb{P}^2 \times \mathbb{P}^3$. Each of the other columns denotes the multi-degree of one polynomial in these projective spaces, and the zeros of these polynomials define hypersurfaces in the projective spaces. To write the polynomials explicitly, we use the following definitions: $\mathbb{P}^1$ is denoted with x^a, were $a = 0, 1$, the $\mathbb{P}^2$ coordinates by y^i with $i = 0, 1, 2$, and for $\mathbb{P}^3$ we have z^m with $m = 0, 1, 2, 3$. Then, the two polynomials are

$$\begin{aligned} p_1 &= \sum_{\substack{a=0,1\\ i=0,1,2}} c_{ai} x^a y^i = c_{00} x^0 y^0 + c_{01} x^0 y^1 + c_{02} x^0 y^2 + c_{10} x^1 y^0 + c_{11} x^1 y^1 + c_{12} x^1 y^2 \ , \\ p_2 &= \sum_{\substack{a=0,1\\ i,j=0,1,2\\ m,\ldots,q=0,\ldots,3}} d_{aijmnpq} x^a y^i y^j z^m z^n z^p z^q \ , \end{aligned} \tag{4.10}$$

where c_{ai} and $d_{aijmnpq}$ are complex coefficients. Therefore, the configuration matrices themselves do not describe explicit spaces but rather families of projective spaces, which share some features. The strength of the notation of the configuration matrices is that the most important features are independent of the prefactors, so for example the Euler numbers, the Hodge number, intersection numbers etc. Therefore, we are only interested in finding equivalent configuration matrices than looking on explicit polynomials. (Note that the configuration matrices are independently of permutations of columns (except the first one) and rows.)

There are various constraints on the coefficients of such a configuration matrix: To be a Calabi-Yau manifold each row has to satisfy the relation

$$n + 1 = \sum_{\alpha} q_n^{\alpha} \ , \tag{4.11}$$

where we use the general description of the configuration matrix as

$$\left[\begin{array}{c|ccc} n_1 & q_1^1 & \ldots & q_K^1 \\ \vdots & \vdots & \ddots & \vdots \\ n_m & q_1^m & \ldots & q_K^m \end{array}\right] . \tag{4.12}$$

The condition (4.11) is equivalent to a vanishing second chern class and therefore crucial for any Complete Intersection Calabi-Yau manifold.

Additionally, we want to fix the dimensionality of our manifold to d dimensions. To reduce the number of freedom given by the sum over the dimensions of the projective spaces we find the constrain:

$$\sum_r n_r = k + d \ , \tag{4.13}$$

where k denotes the number of equations. In [49], it was shown that there can only exist a finite number of independent configuration matrices within this set of rules. Additionally to those constraints, there exists a set of identities which link different configuration matrices to each other. These connected CICYs are equivalent to each other and therefore, they have to be singled out when classifying the CICYs. A summary of the identities, along with explanations, can be found in [47]. In our problem setting we use them as symmetry transformations our method has to find.

Identities – discrete symmetries

As stated above, the simplest transformation which leaves the manifold itself invariant are permutations of rows and columns. Additionally, there are five identities where it was mathematically proven that they leave the manifold invariant. These five identities are:

$$\begin{gathered}
\left[\begin{array}{c|cc} 2 & 2 & \mathbf{a} \\ \mathbf{n} & 0 & \mathbf{q} \end{array}\right] = \left[\begin{array}{c|c} 1 & 2\mathbf{a} \\ \mathbf{n} & \mathbf{q} \end{array}\right], \qquad
\left[\begin{array}{c|cc} 1 & 1 & \mathbf{a} \\ 1 & 1 & \mathbf{b} \\ \mathbf{n} & 0 & \mathbf{q} \end{array}\right] = \left[\begin{array}{c|c} 1 & \mathbf{a}+\mathbf{b} \\ \mathbf{n} & \mathbf{q} \end{array}\right], \\
\left[\begin{array}{c|cc} 3 & 2 & \mathbf{c} \\ \mathbf{n} & 0 & \mathbf{q} \end{array}\right] = \left[\begin{array}{c|c} 1 & \mathbf{c} \\ 1 & \mathbf{c} \\ \mathbf{n} & \mathbf{q} \end{array}\right], \qquad
\left[\begin{array}{c|cc} 1 & 2 & 0 \\ 2 & 1 & \mathbf{c} \\ \mathbf{n} & 0 & \mathbf{q} \end{array}\right] = \left[\begin{array}{c|c} 1 & \mathbf{c} \\ 1 & \mathbf{c} \\ \mathbf{n} & \mathbf{q} \end{array}\right], \\
\left[\begin{array}{c|ccc} 2 & 2 & 1 & 0 \\ 2 & 1 & 1 & \mathbf{a} \\ \mathbf{n} & 0 & 0 & \mathbf{q} \end{array}\right] = \left[\begin{array}{c|cc} 1 & 2 & 0 \\ 2 & 2 & \mathbf{a} \\ \mathbf{n} & 0 & \mathbf{q} \end{array}\right].
\end{gathered} \tag{4.14}$$

The vectors $\mathbf{a}$ and $\mathbf{b}$ are vectors, with zeros everywhere, expect for one entry, the vectors $\mathbf{n}$ and $\mathbf{c}$ are arbitrary vectors, whereas $\mathbf{q}$ is a matrix. Note, that these vectors and matrices have to be chosen in an appropriate way to fulfill the defining properties of a configuration matrix in d dimensions.

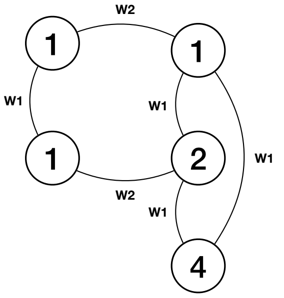
CICYs as graphs

To simplify the training process we used a permutation invariant representation of those configuration matrices. Therefore, we use a graph representation of the configuration matrices. Using graph representations has shown great success in many applications of machine learning, so for example in classifying properties of moleculs (c.f. [50]). Here, we are going to use a simpler approach: We rewrite each configuration matrix as a graph, where we mapped the right side of the configuration matrix to a graph (which is sufficient to reconstruct the whole matrix). In Figure 4.3, there is an example of such a map. We assign different weights to the connections corresponding to rows and columns to make them distinguishable. After this step, we have a representation of the CICYs which is invariant under permutation. As next step, we have to manipulate the data in such a way that we can feed them into the neural network. Therefore, we look at the different

Configuration matrix

$$\left[\begin{array}{c|cc} 1 & 1 & 1 \\ 2 & 1 & 2 \\ 3 & 0 & 4 \end{array}\right]$$

Graph representation



Next neighbors

vertex	horizontal	vertical
1	1	1
1	1	2, 4
1	2	1
2	1	1, 4
4	–	1, 2

Figure 4.3: Example for one CICY and the different representation we used. **Left:** The configuration matrix of the CICY. **Middle:** A graph visualisation of the CICY. Note the different weights for horizontal and vertical edges. **Right:** The nearest neighbours of each node of the the graph in the middle.

types of nodes, how they are connected and what kind of neighbors they have. This gives us 285 different types of nodes, and therefore, we can construct a dictionary of types of nodes. Now, we can construct a 285 dimensional feature vector for each configuration matrix. Additionally, we compute the eigenvalues of the adjacency matrix of the graph for each configuration matrix which leads to an additional 30 dimensional vector (for smaller matrices we padded the vectors with zeros. These information are together sufficient as input for the neural network.[1]

Training of the network

For the neural networks, we need to find a suitable training task which can be computed in a straightforward way from the configuration matrix. We use the topological invariants $h^{1,1}$ and $h^{1,2}$ which were obtained in [51]. From here on, we can pursuit just as in the previous examples: We start constructing 500 representatives of each class by randomly applying permutations and identities (4.14) to the known representatives of the different classes. As next step, we compute the feature vector and the the eigenvalues of the adjacency matrix to get the 315-dimensional input vector. Note that the number of possible representatives might differ, depending on the number of identities applicable. In general, we obtain between 100 and 300 different representatives per configuration matrices, but there are examples where only one representative exists (e.g. the so called quintic hypersurface). The structure of the neural network can be found in Table 4.1, it is a multi-layer perceptron with two distinct output layers which enables the neural network to learn both Hodge numbers. (This is analogous to the example of the superpotential where we learned the classes for the real and the imaginary part.) As in the cases before, we trained the neural network with an Adam-optimizer and a categorical crossentropy loss. After reaching a training accurancy above 95 percent we stopped the training process.

---

[1] Nowadays, we would be able to feed the graph directly to the neural network, but at the time of the project it was unfeasible to construct a graph neural network because no accessible framework existed.

Type	Dimension	Activation	Initializer	Regularization
Input	315			
Dense	315	ReLU	`glorot_uniform`	
Dense	315	ReLU	`glorot_uniform`	l2(10^{-5})
BatchNormalization				
Dense	100	ReLU	`glorot_uniform`	
Dense	100		`glorot_uniform`	l2(10^{-3})
Output 1: Dense	102	softmax		
Output 2: Dense	20	softmax		

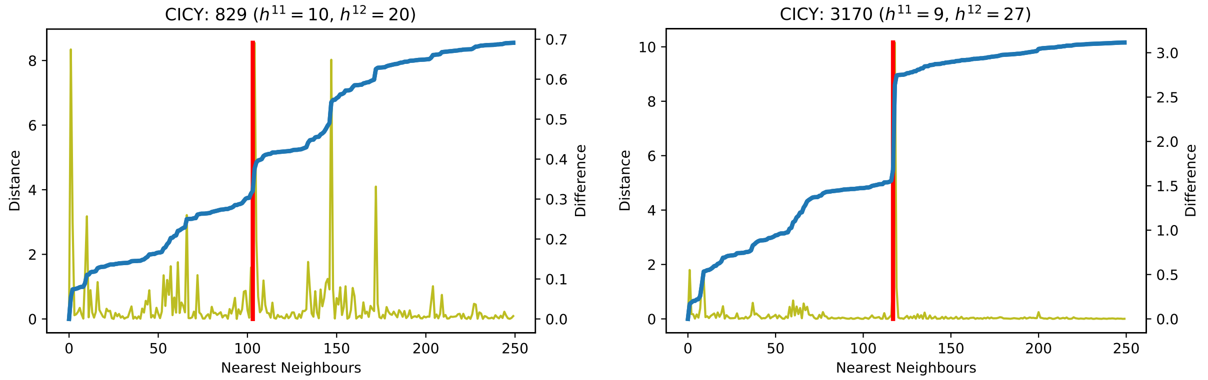
Table 4.1: The architecture of the neural network architecture for Hodge number classification. The last layer before the output layer is called the embedding layer.

Analysis of the results

For the analysis, we only use distinct representatives of each data point. This leaves us with $\sim$ 600,000 data points to compare. As we have to compare 7890 different classes it is unfeasible to analyze the structures using t-SNE. Instead, we analyze the distance of the nearest neighbors in the embedding layer.

For each data point, we find the 250 nearest neighbors (with respect to their Euclidean distance). All of these plots of these distances show a similar structures. We presented two examples in the plots in Figure 4.4. Those lines always have several plateaus in the distance curve and several big jumps between two points which are shown in yellow. We are going to use the biggest jump as a threshold, which appears as a natural choice to us. CICYs which are closer to the original data point are assumed to be of the same class as the original data point. Our requirement is that we always at least consider the closest point as a neighbor. This prediction is pretty successful, given the fact that we are only able to decide whether two configuration matrices are the same by comparing them with the full list of classified CICYs. We want to point out the fact that the neural network is only trained on the Hodge numbers (which are often the same), and not on the class labels of the matrices. A quantitative analysis of the performance can be found in Figure 4.5: We can see that the method is able to correctly classify most of the CICYs correctly. We reach an accuracy rate above 95 percent for 86.6 percent of the CICYs without any additional input, for example, that we do not removed any data points which are prone to fail (such as the quintic) or used any other biases we would normally use.

As next step, we analyze only data points which share the same Hodge number $h^{1,1} = 10$ and $h^{1,2} = 20$, there are 292 distinct CICYs. The analysis works in this regime as well, as one can see on the right side of Figure 4.5: The majority of CICYs – 80.6 percent – is correctly classified. Such a small drop of performance is expected because the complete dataset consists of many classes where the combination of the Hodge numbers is unique, and therefore, the presented method should perfectly work for those CICYs.



$$\left[\begin{array}{c|cccccccccc}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
3 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0
\end{array}\right]
\qquad
\left[\begin{array}{c|cccccccccc}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\
2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
2 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
2 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right]$$

Figure 4.4: Two examples of the method used to define a cluster in the embedding space. We plot in blue the Euclidean distance in the embedding layer of the 250 nearest neighbours to the two fixed CICYs displayed below. The difference between the distances for the points i and $i + 1$ is plotted in yellow. The red line highlights the largest difference and defines the threshold for the cluster. Below are the configuration matrices for the representatives of the CICYs as they have been classified.

Interpretation

To ensure that two CICYs are mathematically equivalent, we have to compare several topological quantities, such as Euler number and the Hodge number, as well as the intersection number. Due to the basis dependence of the intersection number, we cannot tell whether two CICYs are distinct, unless we compare it to the full list of classified CICYs. It is unclear how the neural network is able to map different representatives of the same class together in the embedding space. A deeper understanding would help to understand Wall's theorem [48], and finally lead to a calculation rule how to construct a sufficient basis for each CICY.

4.2 Finding Generators

After we presented a way how to detect the existence of a symmetry, we continue in describing an algorithm how to find the generators and therefore we are able to identify the underlying symmetry. Our starting point is a point cloud, like for example the coordinates

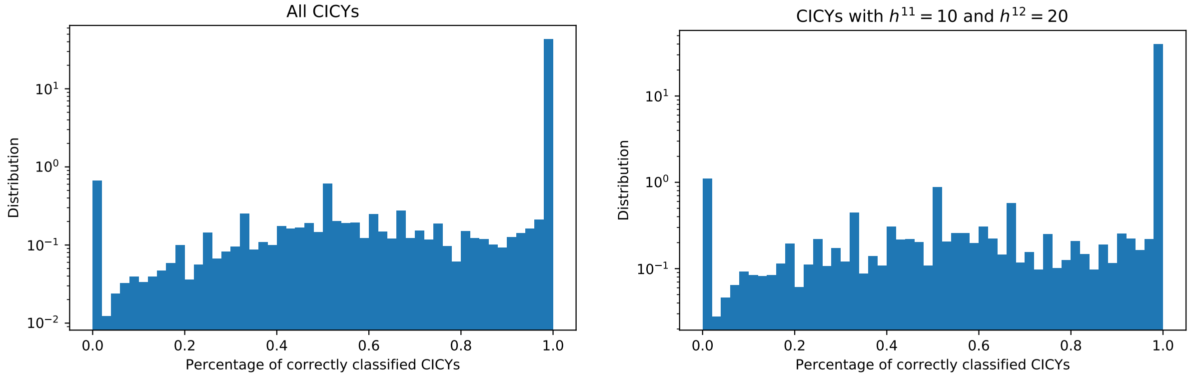


Figure 4.5: Accuracy of our method on the CICY dataset. **Left:** We plotted the distribution of the accuracy for all 686,464 data points. **Right:** Examining the subset of CICYs with Hodge-numbers $h^{1,1} = 10$ and $h^{1,2} = 20$, we find the distribution for the accuracy in the plot on the right. We still perform the analysis of finding nearest neighbours with all data points.

used in the previous examples. We use general considerations how one would usually compute the generators in theoretical settings, and develop an algorithm which is able to identify those generators in points clouds. To demonstrate this method, we construct noisy point clouds in various dimensions and apply the algorithm to these point clouds. After presenting the detected generators, we apply the same method to images and note that a modified method is able to find those in such high dimensional data as well. We conclude with a few remarks on discrete symmetry groups.

4.2.1 Algorithm

The fundamental idea is to apply the methodology used to calculate infinitesimal generators from continuous point cloud P_{con} which is related by some symmetry group and generalize it to discrete point clouds. Two infinitesimal close points p and p' (from the point cloud P_{con}) are linked by a symmetry group as follows

$$p' = p + \epsilon_a T^a p \ , \tag{4.15}$$

where T^a are the generators of the group, while ϵ_a is a vector consisting of small numbers which lead to a connection of the two points. The symmetry group is specified from the generators T^a, therefore, finding them is enough to identify the symmetry group.

The prime example of demonstrating our idea is the case of SO(2), which describes a perfect circle. We take two points pairs on the circle, namely

$$p_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} , \quad p_1' = \begin{pmatrix} 1 \\ \epsilon \end{pmatrix} \quad \text{and} \quad p_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} , \quad p_2' = \begin{pmatrix} -\epsilon \\ 1 \end{pmatrix} , \tag{4.16}$$

which are enough to completely restrict the generator. Applying the equation (4.15) to those points we get an equation system and we find for the generator

$$T^1 = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} . \tag{4.17}$$

This can be identified as the generator of SO(2). It is important to note that there are several advantages working with a continuous point cloud compared to a discrete point cloud:

- We are able to choose points which are in the same plane.
- We know a suitable basis to choose the correct points.
- We have chosen the points in such a way that the orientations of the transformation is correct.
- Compared to a discrete point cloud we have an infinite number of data points, which can be arbitrary close to each other.

Those points are the main reasons for complications when defining the algorithm for a discrete point cloud. The goal is to take a suitable subset of points and their neighbors which all lie in a plane. Then identify the correct orientation of the point pairs and use them to constrain the generator of the subset. After finding one specific generator, we repeat this process multiple times. It is more appropriate to look for arbitrary generators rather then orthogonal ones. As last step, we find the "basis" for the generators which then allows us to identify the symmetry group itself. The algorithm for a discrete point cloud P itself is described in [22] in the following way:

1. As first step, we can remove redundant dimensions and center the dataset to the origin. For this pre-processing we can for example use principle component analysis (PCA). This can be also used to perform a dimensional reduction, and gives you information on how many dimensions are necessary to keep.
2. As next step, we generate an orthonormal basis $(b_1, \ldots, b_n)$: We pick two arbitrary points $p_1, p_2 \in P$ from the dataset. Then, we normalize the vector p_1 and use the normalized vector as first basis vector $b_1 = p_1/||p_1||$. The second basis vector is in the direction of the orthogonal components of the vector p_2, given by the normalized version of vector $p_2 - (p_2 \cdot b_1)b_1$. Then, we can add arbitrary orthonormal vectors to complete the basis of the space.
3. We have to filter out points which are close to the hyperplane H spanned by b_1 and b_2. We want to find the generator which acts in this hyperplane. As condition we use
$$|p \cdot b_i| < \delta \quad \text{for } 2 < i \leq n \ , \tag{4.18}$$

where $\delta > 0$ is a hyperparameter we have to choose by hand. The larger δ the more points are included, but then it also becomes harder to find appropriate generators. Note that in this 'thick' hyperplane points might possess also neighbors which are orthogonal to the plane of interest. Those contribution are removed using condition (4.22).

4. As next step, we want to find point pairs $p, p' \in H$ which are close enough to each other to be usable as neighbors. The condition we use is

$$||p - p'|| < \epsilon \;\text{ for } \forall\, p, p' \in H \;, \tag{4.19}$$

with $\epsilon > 0$ being a hyperparameter as well. Note that we do not only include next neighbors but also other points which are close enough.

5. These point pairs (p, p') are now used to find constraints on the generators in equation (4.15). This equation becomes

$$p' - p = \frac{\sigma_H(p, p')}{\|p\|} \; \|p' - p\| \; Tp \;, \tag{4.20}$$

where T denotes the generator we determine. The distance $\|p' - p\|$ is the same as the ϵ_a in equation (4.15). The normalization factor $1/\|p\|$ generalizes the results to point clouds with points p where $\|p\| \neq 1$. $\sigma_H(p, p')$ denotes the correct direction between the points (p, p') within the hyperplane. It is defined as:

$$\sigma_H(p, p') = \;\text{sign}\left((p \cdot b_1)(p' \cdot b_2) - (p \cdot b_2)(p' \cdot b_1)\right) . \tag{4.21}$$

All components of T which are perpendicular to the hyperplane are constrained using

$$T\; b_i = 0 \;\text{ for } i > 2 \;. \tag{4.22}$$

This condition removes also false directional information arising from point pairs which would actually point in the wrong direction (perpendicular to the hyperplane).

6. The equations (4.21) and (4.22) give us a well defined system of equations, we can solve using linear regression. Note that we weigh the constraints (4.22) higher then the constraints from equation (4.21), and do not enforce it as a hard condition.

7. By applying the steps 2-5 multiple times, we find different generators for 'all' present directional combinations. To filter out the main components of this generator space, we can apply a PCA on all generators. As next step, we analyze the standard deviation in these components. This gives us the right number of present generators of the underlying dataset. Additionally, the PCA also gives us also a basis for the generator space, allowing us to identify the underlying symmetry group.

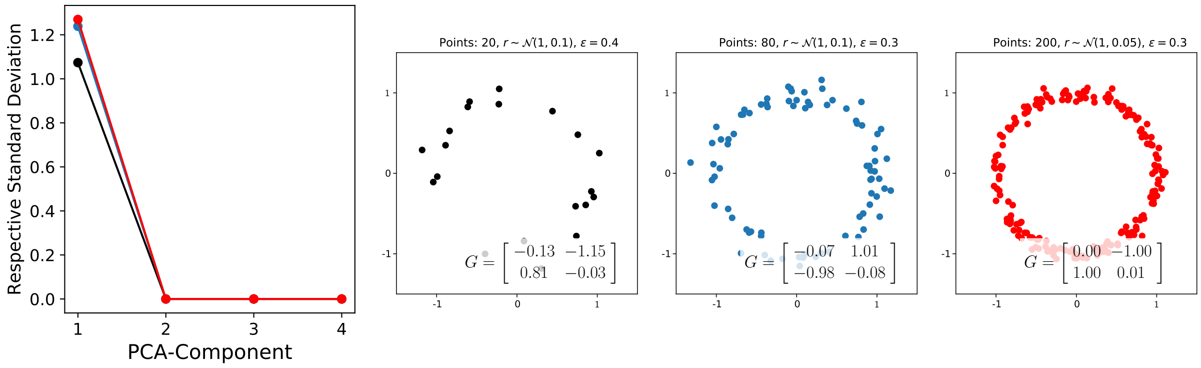


Figure 4.6: We show three examples of point clouds for SO(2). To display the performance of the algorithm for different difficulties, we vary the number of points and use different noise. The respective parameters are shown in the plot title. In the scatter plots, we also display the generator corresponding to the first PCA component which is found by our algorithm.

8. Finally, we have to make a remark on distinguishing unitary from orthogonal groups, so for example between SU(2) and SO(4). The problem is that the orbits are not necessarily distinguishable: Considering the orbit of a point on a unit sphere S^3, it might by generated by both symmetries, because every point of the sphere S^3 can be mapped to the point $(1, 0, 0, 0)$, with SU(2) as well as SO(4) transformations. To distinguish them, we always have to use a pair of point pairs to see a difference between the two transformations. In realistic physical situations, we usually have such point pairs (for example the SU(2) superpotential in section 4.1.1) and we can use these point pairs to distinguish between the symmetry groups. To use the upper example of an S^3: When an SU(2) symmetry is present all generators are fixed when one of the points is moved to the north pole $(1, 0, 0, 0)$, whereas when we use an SO(4) symmetry half of the generators is not fixed yet.

Let us now discuss several experiments for this algorithm. Here, we mainly focus on SO(n) as well as SU(2), but also other symmetry groups are suitable for this algorithm.

4.2.2 Examples

We start our experiments with the most simple example: the already known SO(2). Here, we perform several stability checks and vary the number of points as well as the noise. We construct our noisy point clouds in the following way: We randomly sample distributed points on the circle with radius $r = \mathcal{N}(1, \sigma)$. Results are shown in Figure 4.6. Even for a small point clouds which one would not necessarily identify as a circle the algorithm is able to identify a matrix which is recognizable as the generator of SO(2). The analysis is performed with $\delta = 0.5$. Note the large difference in the standard deviation of the PCA. This clearly indicates that only one generator is present. This holds even in the case with only 20 data points. We continue with higher dimensional problems.

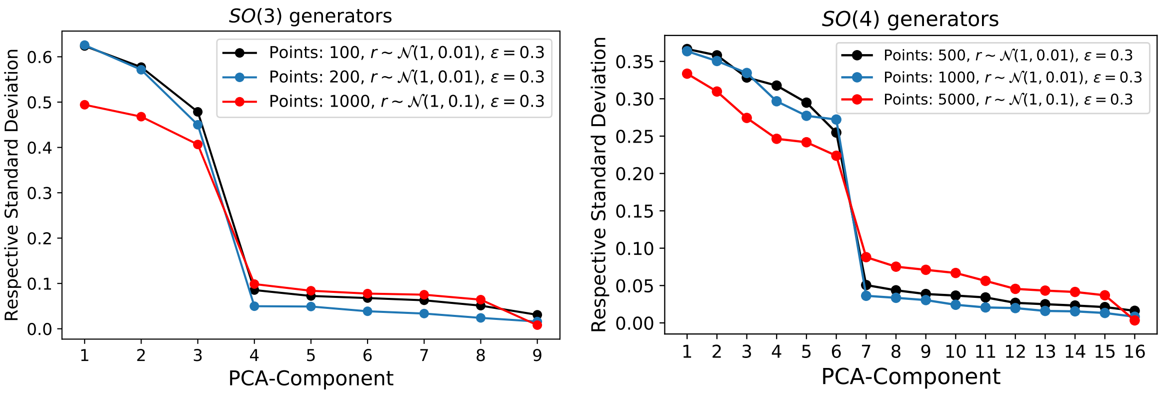


Figure 4.7: **Left:** We display the standard deviation of the PCA components for different examples of SO(3) point clouds. **Right:** The standard deviation of the PCA components of the SO(4) example.

As next steps, we discuss examples in three and four dimensions. We use our algorithm on different point clouds modeling a noisy S^3 and S^4. The results for SO(3) and SO(4) are displayed in figure 4.7. In both setups, we see a steep drop of the standard deviation of the PCA independently of the specific choice of hyperparameters. Note that with increasing dimensions the number of points needed increases because the hyperplane used becomes relatively smaller compared to the total volume of S^d. In three dimensions, we find three generators whereas in four dimensions, we find six. Our numerical results for the generators of SO(3) using 1000 points (red curve in Figure 4.7) are:

$$\begin{aligned} G_1 &= \begin{pmatrix} -0.00 & 0.04 & 0.59 \\ -0.06 & 0.01 & 0.78 \\ -0.59 & -0.82 & -0.01 \end{pmatrix}, \quad G_2 = \begin{pmatrix} -0.01 & -0.98 & -0.13 \\ 0.98 & 0.04 & 0.14 \\ 0.16 & -0.18 & 0.01 \end{pmatrix}, \\ G_3 &= \begin{pmatrix} 0.00 & 0.21 & -0.81 \\ -0.21 & 0.00 & 0.55 \\ 0.78 & -0.61 & -0.03 \end{pmatrix}. \end{aligned} \tag{4.23}$$

From the red curve in four dimensions, we get for the generators of SO(4) the formulae

$$
\begin{aligned}
G_1 &= \begin{pmatrix} 0.02 & 0.50 & -0.11 & 0.25 \\ -0.52 & -0.00 & 0.39 & 0.60 \\ 0.10 & -0.41 & -0.00 & -0.38 \\ -0.28 & -0.59 & 0.38 & -0.02 \end{pmatrix}, \; G_2 = \begin{pmatrix} 0.00 & 0.08 & 0.41 & -0.07 \\ -0.09 & -0.00 & -0.31 & -0.29 \\ -0.48 & 0.24 & -0.02 & -0.78 \\ 0.06 & 0.29 & 0.81 & 0.02 \end{pmatrix}, \\
G_3 &= \begin{pmatrix} 0.02 & 0.13 & 0.42 & -0.29 \\ -0.09 & 0.04 & 0.78 & -0.32 \\ -0.45 & -0.76 & -0.02 & 0.13 \\ 0.31 & 0.33 & -0.13 & -0.03 \end{pmatrix}, \; G_4 = \begin{pmatrix} 0.03 & 0.55 & 0.50 & 0.42 \\ -0.57 & -0.00 & -0.30 & -0.08 \\ -0.45 & 0.31 & 0.02 & 0.44 \\ -0.40 & 0.12 & -0.43 & -0.03 \end{pmatrix}, \\
G_5 &= \begin{pmatrix} 0.01 & 0.63 & -0.48 & -0.50 \\ -0.64 & 0.01 & -0.16 & -0.29 \\ 0.48 & 0.14 & -0.00 & 0.03 \\ 0.51 & 0.32 & 0.02 & -0.01 \end{pmatrix}, \; G_6 = \begin{pmatrix} -0.02 & -0.01 & -0.37 & 0.62 \\ 0.01 & 0.02 & 0.24 & -0.61 \\ 0.40 & -0.24 & -0.03 & -0.13 \\ -0.67 & 0.59 & 0.14 & 0.02 \end{pmatrix}.
\end{aligned}
\tag{4.24}
$$

For the next steps, we focus on the four dimensional space and discuss subgroups of the SO(4), namely SU(2) and SO(2) × SO(2). Our method is able to identify the subset of generators of SO(4) which defines the searched symmetry group. Again we tested our method on different hyperparameters. The results can be found in Figure 4.8. For SU(2), we find the following generators for the red curve (5000 points):

$$
\begin{aligned}
G_1 &= \begin{pmatrix} -0.01 & 0.52 & 0.47 & -0.11 \\ -0.52 & 0.00 & 0.08 & 0.49 \\ -0.47 & -0.08 & 0.01 & -0.50 \\ 0.12 & -0.48 & 0.50 & 0.00 \end{pmatrix}, \; G_2 = \begin{pmatrix} -0.00 & -0.24 & 0.43 & 0.46 \\ 0.26 & 0.00 & -0.52 & 0.39 \\ -0.43 & 0.51 & -0.00 & 0.35 \\ -0.45 & -0.39 & -0.34 & -0.01 \end{pmatrix}, \\
G_3 &= \begin{pmatrix} 0.00 & -0.39 & 0.30 & -0.50 \\ 0.37 & 0.01 & 0.51 & 0.32 \\ -0.31 & -0.50 & 0.01 & 0.39 \\ 0.49 & -0.31 & -0.40 & 0.00 \end{pmatrix}.
\end{aligned}
\tag{4.25}
$$

Note that it was necessary to use point pairs (x, y) to distinguish SU(2) from SO(4). This is described in bullet 8 in section 4.2.1 in more detail. For SO(2) × SO(2) we find for 100 points (black curve in Figure 4.8) the generators:

$$
G_1 = \begin{pmatrix} -0.03 & 0.14 & 0.03 & -0.01 \\ -0.31 & 0.01 & 0.03 & 0.01 \\ 0.04 & 0.01 & -0.01 & 0.95 \\ -0.1 & -0.06 & -0.98 & 0.04 \end{pmatrix}, \quad G_2 = \begin{pmatrix} 0.00 & -1.13 & 0.09 & 0.01 \\ 0.78 & -0.04 & -0.03 & -0.03 \\ 0.03 & 0.04 & -0.02 & 0.19 \\ 0.08 & -0.02 & -0.23 & -0.00 \end{pmatrix}. \tag{4.26}
$$

4.2.3 Applications to the Mexican hat Potential and the Superpotential

To connect the algorithm to section 4.1, we apply our presented algorithm to the point clouds resulting from the analysis of the embedding layer for both problems, the Mexican hat potential (4.4) and the superpotential (4.6).

For the Mexican hat potential we use the radius $r = 1.45$. This class has 497 point clouds and can be seen in Figure 4.9 on the left. This point cloud has nearly no noise and

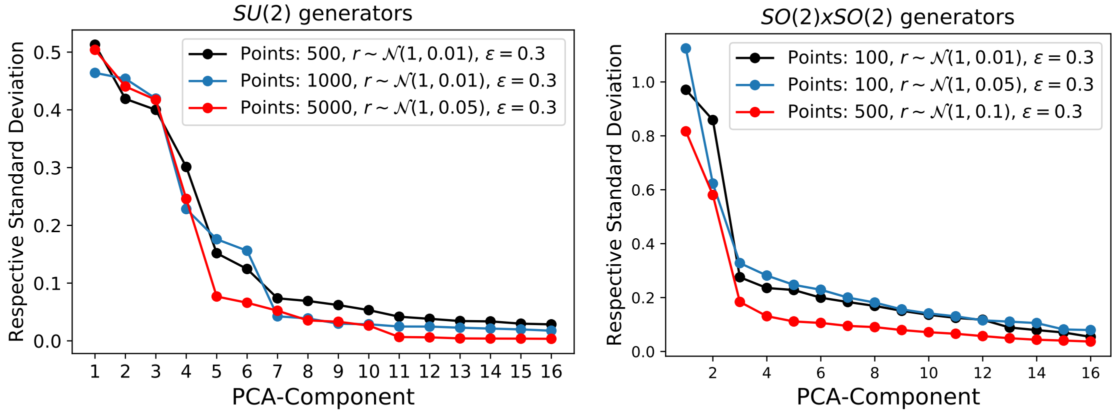


Figure 4.8: **Left:** We display the standard deviation of the PCA components for different examples of SU(2) point clouds. **Right:** The standard deviation of the PCA components of the SO(2) × SO(2) example.

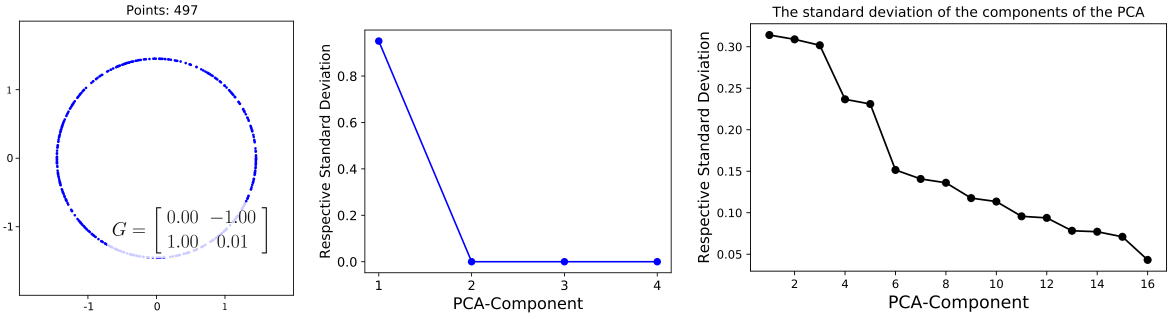


Figure 4.9: The analysis for the Mexican hat potential in section 4.1.1. **Left:** The point-cloud found in section 4.1.1. **Middle:** The standard deviation of the PCA components of the Mexican hat example. **Right:** The standard deviation of the PCA components of the superpotential example.

therefore, the algorithm is able to find one dominant generator for this data set (see Figure 4.9 in the middle).

We proceed with the superpotential and apply the same analysis here. In contrast to the Mexican hat potential, we use every class (each with ~ 1000 representatives) found by our analysis of the embedding layer and apply the described algorithm on these clases. The results can be found in Figure 4.9. As we can see, we get the same structure as for the SU(2) point clouds. Therefore, we conclude that the combination of both parts gives us a good method to find the underlying symmetries of a potential.

4.2.4 Rotated MNIST

The final application of our algorithm is connected to the well-known dataset of handwritten digits MNIST. The goal is to rotate the images and find the generator of SO(2).

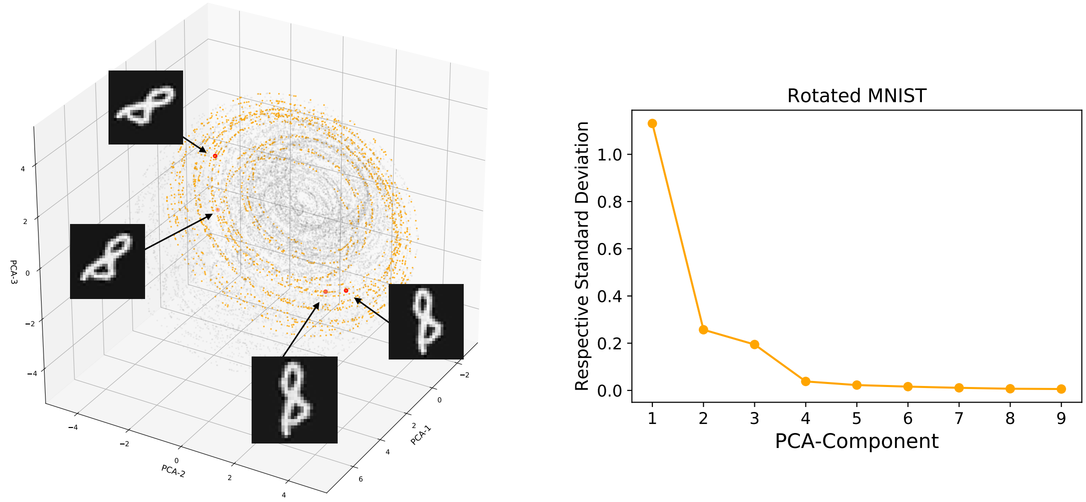


Figure 4.10: **Left:** Point cloud of the first three PCA components of the rotated MNIST dataset. In orange, we highlighted the orbits of multiple images of the digit eight. Gray points correspond to the other digits. **Right:** The results of the standard deviation for the generators for the digit eight. Those have been identified from the point cloud.

Therefore, we apply rotations on the 28×28 images. However, due to the size of this space (784 dimensions), we first apply a PCA on the dataset before we apply our algorithm and identify the generator.

Let us describe the approach in detail:

1. We only pick MNIST-images of one digit, in our example the 8 and pick the first 20 images of it.
2. We rotate the images with random angles to get 2000 rotated versions in total of different "eights".
3. Applying PCA on the 784 dimensional space, we end up with a point cloud in three dimensions. A visualization can be found in Figure 4.10 .
4. On this space, we apply our algorithm described in section 4.2.1 to the point cloud.

The algorithm finds as the dominant generator (see Figure 4.10 for the standard deviation) of this point cloud the generator of SO(2) given by

$$G = \begin{pmatrix} -0.06 & -0.00 & -0.07 \\ 0.01 & -0.01 & 1.00 \\ 0.08 & -0.99 & 0.04 \end{pmatrix} . \tag{4.27}$$

4.2.5 Discrete symmetries – CICYs

As last step, let us briefly comment on the example of CICYs discussed in Section 4.1.2. For discrete symmetries our algorithm described in the beginning of this section, is not applicable because it is based on infinitesimal close point pairs. To identify underlying symmetries here, we need to find identical transformations in different orbits on the input space and recognize them as the same ones. Due to the construction of the dataset using exactly such identities and the clustering in the embedding layer described in section 4.1.2, our network is able to detect those symmetries. However, the search for possible new identities within the dataset was beyond the scope of the project [22].

Chapter 5

Symmetry Control Neural Network

In this chapter, we describe a method how to find conserved quantities using machine learning purely from observations of the trajectories of particles. As first step, we review the construction of the Hamiltonian Neural Network (HNN) presented by Greydanus et al. [52]. As second step, we utilize this idea to find the conserved quantities of trajectories purely using data from observation. In the second section, we demonstrate experimentally the success of the idea of the Symmetry Control Neural Network (SCNN). As final step, we propose an application for physical problems with known Hamiltonians. The main findings of this chapter are based on the results in [23].

5.1 Reviewing Hamiltonian Neural Networks

In many problems in machine learning we have to take physical laws into account, whether it is analyzing video sequences, training an reinforcement agent, or predicting the future development of a system. One of the keys to achieve such good performances on physical tasks is the use of suitable constructed neural networks. Therefore, the neural network must be able to predict the changes of the investigated system. One way to predict the change of the system is to compute the new phase space variables $(\mathbf{q}_t, \mathbf{p}_t)$ after a finite time step Δ. This is given by

$$f_\theta : \quad (\mathbf{q}_t, \mathbf{p}_t) \longleftrightarrow (\mathbf{q}_{t+\Delta t}, \mathbf{p}_{t+\Delta t}) \ , \tag{5.1}$$

where f_θ is a function parameterized by trainable weights θ. It turns out that while this is sufficient for simulation tasks [53] with a large number of particles, this does not provide any information about conserved quantities or the physics behind the problem. For physics, it turns out it is more suitable to predict $\left(\frac{d\mathbf{q}}{dt}, \frac{d\mathbf{p}}{dt}\right) \equiv (\dot{\mathbf{q}}, \dot{\mathbf{p}})$ using position and momenta $(\mathbf{q}, \mathbf{p})$ as input, so:

$$f_\theta : \quad (\mathbf{q}, \mathbf{p}) \longleftrightarrow (\dot{\mathbf{q}}, \dot{\mathbf{p}}) \ , \tag{5.2}$$

where f_θ is a function parameterized by trainable weights θ. This training can be done in a completly supervised fashion, training the network to predict $(\dot{\mathbf{q}}, \dot{\mathbf{p}})$ directly. Such an approach, combined with so called graph neural networks was used in [54] to find analytical expressions of forces between particles.

To predict the state of the system at time t_2 given the state at t_1, we can integrate over the time derivatives:

$$\begin{aligned}(\mathbf{q_{t_2}}, \mathbf{p_{t_2}}) &= (\mathbf{q_{t_1}}, \mathbf{p_{t_1}}) + \int_{t1}^{t_2} \mathrm{d}t \, (\dot{\mathbf{q}}, \dot{\mathbf{p}}) \\ &= (\mathbf{q_{t_1}}, \mathbf{p_{t_1}}) + \int_{t1}^{t_2} \mathrm{d}t f_\theta \left(\mathbf{q_t}, \mathbf{p_t}\right) \, .\end{aligned} \tag{5.3}$$

Using this approach and integrating over a period of time it turns out the neural network is not able to conserve the energy of the system. It actually accumulates small errors leading to the problem that the system gains (or looses) energy. This leads to an unphysical behavior of the system we would like to avoid. The goal is to find an approach where the energy is conserved.

A method to tackle this problem is the Hamiltonian Neural Network (HNN) [52]. The basic idea is to the learn the Hamiltonian directly, and then using Hamilton's equation to find the time derivatives of position and momenta. Therefore, the goal is to learn a scalar function $\mathcal{H}_\phi$ (which is equivalent to the Hamiltonian in classical mechanics)

$$\mathcal{H}_\phi : \quad (\mathbf{q}, \mathbf{p}) = \mathcal{H}_\phi \left(\mathbf{q}, \mathbf{p}\right) \, . \tag{5.4}$$

This scalar function can be used to compute the time derivatives of position and momenta using the equation for time evolution in classical mechanics, given by

$$\frac{d\mathbf{q}}{dt} = \{\mathbf{q}, \mathcal{H}\} = \frac{\partial \mathcal{H}}{d\mathbf{p}} \, , \qquad \frac{d\mathbf{p}}{dt} = \{\mathbf{p}, \mathcal{H}\} = -\frac{\partial \mathcal{H}}{\mathbf{dq}} \, , \tag{5.5}$$

where $\{\bullet, \bullet\}$ are Poisson brackets. They are defined as

$$\{f, g\} := \frac{\partial f}{\partial \mathbf{q}} \frac{\partial g}{\partial \mathbf{p}} - \frac{\partial f}{\partial \mathbf{p}} \frac{\partial g}{\partial \mathbf{q}} \, . \tag{5.6}$$

Using those formulae the Network function $\mathcal{H}_\phi$ can be trained to minimize the loss function

$$\mathcal{L}_{\mathrm{HNN}} = \left\| \frac{\partial \mathcal{H}_\phi}{\partial \mathbf{p}} - \frac{d\mathbf{q}}{dt} \right\|_2 + \left\| \frac{\partial \mathcal{H}_\phi}{\partial \mathbf{q}} + \frac{d\mathbf{p}}{dt} \right\|_2 \, . \tag{5.7}$$

The function $\mathcal{H}_\theta \left(\mathbf{q}, \mathbf{p}\right)$ can be modeled with a multi-layer perceptron giving only one single output value, the derivations are computed using auto-differentiation. As a result, the neural network returns a conserved quantity, a scalar, equivalent to the energy of the system. Using the data points given to train the neural network, we can fit an analytical function to this output function. As a result for a given dataset, we get he Hamiltonian of the underlying physical system. For experimental results and more explanations on the method, we refer the interested reader to the paper by Greydanus et al. [52].

As a next step, we move on to the Symmetry Control Neural Network (SCNN), which enables us to learn conserved quantities, and as a consequence, symmetries directly [23]. Here we use the predictions of the HNN as benchmark for the SCNN to compare the two directly.

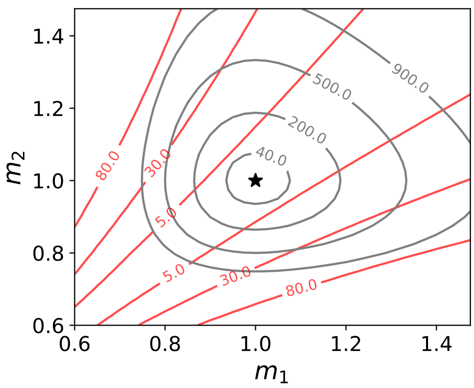


Figure 5.1: **Effect of the additional components of the loss function:** In gray, we show the contours for the HNN-loss (5.11), and in red the contours of the Poisson loss (5.12) using the angular momentum $\|\{L, H\}\|_2$) in the 2-body Hamiltonian (5.15). Here we used the masses m_1 and m_2 of the two bodies as parameters of the system. The data for the model corresponds to $m_1 = m_2 = g = 1$ (indicated with a star). We evaluate the loss for our training set. The analytical constraints from evaluating the Poisson bracket $\{L, H\} \sim (m_1 - m_2)$ is clearly visible as line for $m_1 = m_2$. This provides an additional constraint on the parameters of the model.

5.2 Theory of the Symmetry Control Neural Network

Let us start by briefly defining the notation of Hamiltonian mechanics. We consider a classical system with N particles in d spatial dimensions. Such a system can be described by the variables $(\mathbf{q}, \mathbf{p})$, where $\mathbf{q} = (q_1, ..., q_{N \cdot d})$ are typically the position coordinates for each dimension of the particles and $\mathbf{p} = (p_1, ..., p_{N \cdot d})$ are the corresponding momenta. As mentioned in the previous chapter, we want to learn the time evolution of our system and are therefore interested in $(\dot{\mathbf{q}}, \dot{\mathbf{p}})$.

Additionally, we want to find the conserved quantities of the system for two reasons:

1. It was shown in [55] that utilizing conserved quantities of a Hamiltonian improves the performance and reduces the risk of overfitting. A toy model with only two parameters is described in Figure 5.1. Here, we can see the effect of the constrains on the Hamiltonian.

2. Symmetries lead directly to conserved quantities (c.f. Noether's theorem). Therefore, when learning conserved quantities, we also learn the symmetries of a system. This is interesting from a physical point of view as it allows to examine new systems.

To find the conserved quantities, we utilize the Poisson brackets (5.6), which enable us to compute not only the time evolution of $(\mathbf{q}, \mathbf{p})$, but also the time evolution of all other quantities $g(\mathbf{q},\mathbf{p})$ which does not explicitly depend on time:

$$\frac{dg(\mathbf{q},\mathbf{p})}{dt} = \sum_{i=1}^{N\cdot d} \frac{\partial g}{\partial q_i}\frac{dq_i}{dt} + \frac{\partial g}{\partial p_i}\frac{dp_i}{dt} = \{g,\mathcal{H}\} \ , \tag{5.8}$$

where we have used the Hamiltonian equations (5.5) in the last step. Therefore, a vanishing Poisson bracket ensures, that $g(\mathbf{q},\mathbf{p})$ is a conserved quantity. Additionally, we are going to need further restriction to find different, non-vanishing conserved quantities which differ from each other.

To solve this problem, we are going to make use of cyclic coordinates. Cyclic coordinates are a set of generalized coordinates for which the momenta are conserved. These coordinates still have to fulfill the Poisson algebra as well as Hamiltonian's equation. To find such coordinates, we have to find a suitable diffeomorphic coordinate transformation which leaves the structure of the phase space invariant. These constraints are given by

$$\begin{aligned} &T : (\mathbf{q},\mathbf{p}) \mapsto (\mathbf{Q}(\mathbf{q},\mathbf{p}),\mathbf{P}(\mathbf{q},\mathbf{p}))\,, \\ &\{f,g\}_{\mathbf{p},\mathbf{q}} = \{f,g\}_{\mathbf{P},\mathbf{Q}}\,, \\ &\mathcal{H}(\mathbf{p},\mathbf{q}) = \tilde{\mathcal{H}}(\mathbf{P}(\mathbf{p},\mathbf{q}),\mathbf{Q}(\mathbf{p},\mathbf{q}))\ . \end{aligned} \tag{5.9}$$

We are interested in finding transformations T such that at least one coordinate satisfies

$$0 = \dot{P}_i = -\frac{\partial \mathcal{H}}{\partial Q_i} = \{P_i,\mathcal{H}\}\,. \tag{5.10}$$

Such a coordinate P_i is conserved in the system and the Hamiltonian does not depend on the associated Q_i, i.e. it depends on fewer degrees of freedom and the motion in phase space is restricted to a lower dimensional manifold. Therefore, we do not only learn the conserved quantities, but also restrict the Hamiltonian (referring to Figure 5.1 again) to improve its accuracy.

5.2.1 Structure of the SCNN

As we mentioned before, we are using two different neural networks to find both, the conserved quantities as well as the Hamiltonian (which is a conserved quantity itself).

As first step, we define a network T_ψ which performs the coordinates transformation, and in the second step, using those generalized coordinates as input for the Hamiltonian neural network $\mathcal{H}_\phi$. The structure of those networks can be found in Figure 5.2. We have to translate the necessary conditions into loss functions in order to find the correct structure of the coordinate transformations.

1. The first loss function ensures that our Hamiltonian satisfies Hamiltonian equations (5.5), which we can ensure using

$$\mathcal{L}_{\text{HNN}} = \sum_{i=1}^{N\cdot d} \left\| \frac{\partial \mathcal{H}_\phi(\mathbf{P},\mathbf{Q})}{\partial p_i} - \frac{dq_i}{dt} \right\|_2 + \left\| \frac{\partial \mathcal{H}_\phi(\mathbf{P},\mathbf{Q})}{\partial q_i} + \frac{dp_i}{dt} \right\|_2 \ . \tag{5.11}$$

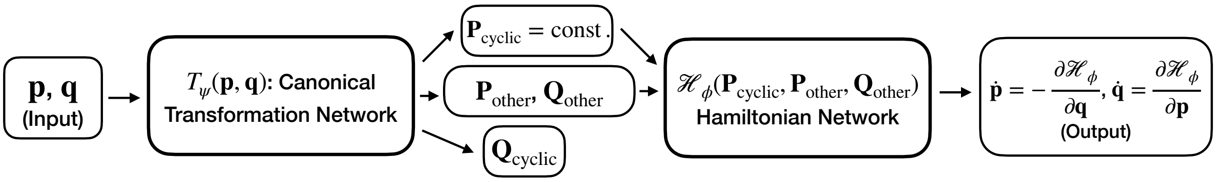


Figure 5.2: The structure and the different parts of Symmetry Control Neural Networks: The network T_ψ corresponds to a coordinate transformation from the input coordinates $(\mathbf{p}, \mathbf{q})$ to other (also canonical) coordinates. Some of those transformed coordinates are forced to be cyclic. This is followed by the Hamiltonian network H_ϕ using the new coordinates as input. The output of the Hamiltonian network corresponds to the Hamiltonian function in theoretical physics and is used to calculate the time derivatives of the initial coordinates.

The time derivatives are provided by the data and the derivatives of the Hamiltonian with respect to the input variables can be obtained using auto-differentiation. This is the same loss function as introduced in [52].

2. To ensure that our transformation T_ψ are of the type we are interested in (cf. Eq. (5.9)), i.e. our new coordinates fulfill the Poisson algebra, we use the loss function

$$\mathcal{L}_{\text{Poisson}} = \sum_{i,j=1}^{N \cdot d} \|\{Q_i, P_j\} - \delta_{ij}\|_2 + \sum_{i,j>i}^{N \cdot d} \|\{P_i, P_j\}\|_2 + \|\{Q_i, Q_j\}\|_2 \ , \tag{5.12}$$

   where in practical applications we only enforce this loss function on n cyclic coordinate pairs. The first part of this loss function ensures that a vanishing solution is not allowed.

3. Hamilton's equations have still to be satisfied with respect to the new coordinates. For the cyclic coordinates we have enforced by the architecture that $\mathcal{H}_\phi$ is independent of Q_i. However, to ensure that P_i is actually conserved, we require the following additional loss function

$$\begin{aligned} \mathcal{L}_{\text{HQP}}^{(n)} =& \sum_{i=1}^{n} \left\| \frac{dP_i}{dt} \right\|_2 + \left\| \frac{dQ_i}{dt} - \frac{\partial \mathcal{H}_\phi(\mathbf{P}, \mathbf{Q})}{\partial P_i} \right\|_2 \\ &+ \beta \sum_{i=n+1}^{N \cdot d} \left\| \frac{dP_i}{dt} + \frac{\partial \mathcal{H}_\phi(\mathbf{P}, \mathbf{Q})}{\partial Q_i} \right\|_2 + \left\| \frac{dQ_i}{dt} - \frac{\partial \mathcal{H}_\phi(\mathbf{P}, \mathbf{Q})}{\partial P_i} \right\|_2 \ , \end{aligned} \tag{5.13}$$

   where n denotes the number of cyclic variables we are imposing and β denotes a hyperparameter for our experiments. For the search for conserved quantities $\beta = 0$ is the best choice, when looking for coordinate transformations we have to choose $\beta \neq 0$ and therefore, restrict the non-cyclic coordinates as well. The time derivatives can be calculated using expressions in (5.8) or using the chain rule.

Our total loss function is a weighted sum of these three components:

$$\mathcal{L} = \mathcal{L}_{\text{HNN}} + \alpha_1 \mathcal{L}_{\text{Poisson}} + \alpha_2 \mathcal{L}^{(n)}_{\text{HQP}} \ , \tag{5.14}$$

where the weights α_i are tuned.

Note that for many physical problems we already know some of the conserved quantities, such as total momentum conservation or angular momentum conservation.

5.2.2 Experiments

In the original paper [23] we used two different types of neural networks to show the success of the Symmetry Control Neural Networks in predicting the time evolution of the system.

- **SCNN-base:** We compare the performance of Symmetry Control Neural Networks with HNNs and baseline neural networks on the accuracy of integrated trajectories for $(\mathbf{p}_0, \mathbf{q}_0)$. We vary the number of conserved quantities up to the maximal allowed number of independent conserved quantities.
- **SCNN-constraint:** We explore whether imposing domain knowledge about symmetries improves the performance. This is motivated by the fact that we often know about the existence of certain conserved quantities we can already include by hand.

Here, we will only present the results of the conserved quantities, as well as the analytical formula of the Hamiltonian, and therefore, we will not take the proper distinction of the difference. For the results of the improvement of the accuracy in simulations, we refer the interested reader to the papers [23, 56]. We tested several examples to show the framework on classical mechanics.

Two-body problem

The first example is the **two-body problem** in two dimensions which is governed by the Hamiltonian

$$\mathcal{H} = \frac{p_{x1}^2}{2m_1} + \frac{p_{y1}^2}{2m_1} + \frac{p_{x2}^2}{2m_2} + \frac{p_{y2}^2}{2m_2} - \frac{g}{\|\mathbf{q}_1 - \mathbf{q}_2\|_2} \ . \tag{5.15}$$

To simplify the problem we set $m_1 = m_2 = g = 1$.

To analyze the conserved quantities, we fit polynomials in $(\mathbf{q}, \mathbf{p})$ to the output of the cyclic coordinates. Picking the solution with the lowest degree and good accuracy[1] we recover known conserved quantities of this system in the **SCNN-base**-model:

$$\begin{aligned} P_{c1} &= -\,4.21\ p_{x1} - 4.21\ p_{x2} - 1.26\ p_{y1} - 1.29\ p_{y2}\ , \\ P_{c2} &= -\,0.93\ p_{x1} - 0.92\ p_{x2} - 3.23\ p_{y1} - 3.22\ p_{y2}\ , \\ L &= -\,1.07\ q_{x1}p_{y1} + 0.88\ q_{x1}p_{y2} + 0.93\ q_{x2}p_{y1} - 1.03\ q_{x2}p_{y2} \\ &\quad + 1.01\ q_{y1}p_{x1} - 0.89\ q_{y1}p_{x2} - 0.92\ q_{y2}p_{x1} + 0.99\ q_{y2}p_{x2}\ . \end{aligned} \tag{5.16}$$

---

[1] We check that there is no significant change in the fit accuracy when including higher order polynomials.

This are accurate fits of the exact conserved quantities

$$\begin{aligned} P_x =& p_{x1} + p_{x2} \,, \\ P_y =& p_{y1} + p_{y2} \,, \\ L =& \left(p_{x1} - p_{x2}\right)\left(q_{y1} - q_{y2}\right) - \left(p_{y1} - p_{y2}\right)\left(q_{x1} - q_{x2}\right) \ . \end{aligned} \tag{5.17}$$

Due to the structure of the Hamiltonian and the energy (which is a $1/r$ function) we have not been able to find the accurate polynomial formula for the Hamiltonian.

Coupled harmonic oscillator

As next system, we are going to look at the **n-dimensional coupled oscillator**. The Hamiltonian of this system is

$$\mathcal{H} = \sum_{i=1}^{n} \frac{p_i^2}{2m_i} + \sum_{i,j=1}^{n} q_i A_{ij} q_j \ , \tag{5.18}$$

where we choose $m_i = 1/2$ and the symmetric $n \times n$ dimensional coupling matrix A_{ij} with positive eigenvalues. We choose a non-diagonal matrix to get a coupling between the particles and therefore, the potential energy of each oscillator depends on the position of the other particles.

We look at the case where $n = 2$. We are able to find the energy of the decoupled sub-system with the Symmetry Control network. The conserved quantities we find are the energies of the two subsystem, e.g.:

$$\begin{aligned} P_1 &= 0.67\ q_1^2 - 3.6\ q_1 q_2 + 8.23\ q_2^2 + 0.46\ p_1^2 - 1.78\ p_1 p_2 + 4.2\ p_2^2 \,, \\ P_2 &= -4.53\ q_1^2 - 1.74\ q_1 q_2 - 0.9\ q_2^2 - 4.5\ p_1^2 - 1.88\ p_1 p_2 - 0.57\ p_2^2 \,. \end{aligned} \tag{5.19}$$

We neglect all terms with factors smaller than 0.05. The analytic formula of the Hamiltonian can also be found accurately given by

$$H = 1.02\ q_1^2 - 0.41\ q_1 q_2 + 1.89\ q_2^2 + 0.97\ p_1^2 + 0.99\ p_2^2 \,, \tag{5.20}$$

neglecting terms smaller than 0.05.

Spherical Pendulum

The next system is the **spherical pendulum** with the Hamiltonian

$$\mathcal{H} = \frac{p_x^2 + p_y^2}{2m} - m\ g\ \sqrt{l^2 - q_x^2 - q_y^2} \ , \tag{5.21}$$

where we use as hyperparameters the mass $m = \frac{1}{2}$, the coupling $g = \frac{1}{2}$ and the length of pendulum $l = 1$. The spherical pendulum has two types of conserved quantities: the angular momentum

$$L = q_x p_y - q_y p_x \ , \tag{5.22}$$

and the energies of the subsystems as it can be viewed as a two-dimensional harmonic oscillator with degenerate eigenvalues.

Using the same architecture as for the harmonic oscillator, we are able to find the following conserved quantities:

$$\begin{aligned} P_1 =& 1.19\ q_x^2 + 0.4\ q_x q_y + 0.72\ q_y^2 + 1.77\ p_x^2 \\ & + 0.51\ p_x p_y + 1.09\ p_y^2 - 0.83\ q_x p_y + 0.84\ q_y p_x\ , \\ P_2 =& -0.65\ q_x^2 + 0.58\ q_x q_y - 1.38\ q_y^2 - 0.96\ p_x^2 \\ & + 0.77\ p_x p_y - 1.92\ p_y^2 - 1.18\ q_x p_y + 1.18\ q_y p_x\ , \end{aligned} \tag{5.23}$$

with MSE of 0.003 and 0.002 and we omit terms below 0.05. Here, we are able to identify the two different subsystems. The analytical formula for the Hamiltonian we found using the SCNN is:

$$H = 0.68\ q_1^2 + 0.71\ q_2^2 + 1.02\ p_1^2 + 1.02\ p_2^2\ . \tag{5.24}$$

When comparing the result to the true Hamiltonian (5.21), we notice that we find a different structure in the formula. An exact comparison shows that the two formulae only differ slightly by a factor of 0.005 using a mean squared error for the comparison.

5.2.3 Application to physics

In physics, the Hamiltonian for many systems is already known. Therefore, we do not have to learn it, but can insert the Hamiltonian itself by hand. As a result, we have to learn only the conserved quantities, and not the Hamiltonian. Therefore, we can neglect the second part of the Symmetry Control Neural Network and only look at the Coordinate Transformation Neural Network. We also have to modify the loss functions. In this version of the SCNN, we are interested in finding n conserved quantities. Therefore, we will only have two terms in the loss function.

1. The first terms are the known Poisson loss. They are necessary to find distinct conserved quantities:

$$\mathcal{L}_{\text{Poisson}} = \sum_{i,j=1}^{n} \|\{Q_i, P_j\} - \delta_{ij}\|_2 + \sum_{i,j>i}^{n} \|\{P_i, P_j\}\|_2 + \|\{Q_i, Q_j\}\|_2\ , \tag{5.25}$$

2. Additionally, the new coordinates still have to fulfill Hamilton's equation, but we cannot directly reuse the third part of the loss function (5.13), because we already have the Hamiltonian given here. A possible modification is:

$$\mathcal{L}^{(n)}_{\text{HQP}} = \sum_{i=1}^{n} \left\| \frac{dP_i}{dt} \right\|_2 + \left\| \frac{dQ_i}{dt} - \{\mathcal{H}, Q_i\} \right\|_2 \tag{5.26}$$

Note, that a pre-computation of the derivatives $\frac{d\mathcal{H}}{dq_i}$ and $\frac{d\mathcal{H}}{dp_i}$ speeds up the calculation. This suggested neural network has not been applied to any physically challenging problem settings and should only point out interesting perspectives for further research in this direction.

For many problems in field theory the down side of this approach is that it can only find a fixed number of conserved quantities. One might be able to raise the number of those conserved quantities, but in field theoretical problem an infinite amount of conserved quantities might occur. For those kinds of problems we will need a different approach to find solution. A possible solution is presented in the next chapter.

Chapter 6

Integrability ex Machina

In this chapter, we are going to discuss a third rather abstract method to find unknown conserved quantities. The goal is to find a mathematical structure called Lax pair. If we are able to find such a structure in a physical system, we know that this system is integrable and therefore, has at least the same amount of conserved quantities as the number of degrees of freedom. Additionally, when we find a Lax pair of a system we are able to compute the conserved quantities of the system. This section is organized as follows: As a first, we give a short introduction into integrability and conditions when a system is integrable. In the second step, we introduce a method how we can translate such problems into an optimization task which can be solved by machine learning algorithms. Third, we are going to apply this algorithm on different systems. We will conclude with examining perturbations of integrable systemd and whether those perturbations are integrable or not. This section is mainly based on the publication [24].

6.1 Review of Integrability

We will start with an overview of the concepts and definitions of integrability. This part is based on the review by Beisert [57]. Another notion of symmetries is the concept of integrability. Roughly speaking, a system is called integrable, if it has at least as many conserved quantities as degrees of freedom. Those conserved quantities have to be in involution, meaning that $\{F_i, F_j\} = 0$ for all conserved quantities F_i and F_j. Typical examples are:

- Harmonic oscillator in d dimensions: Here, the energy of the subsystems are the conserved quantities.

- Two-body problem: It has the energy, the total momentum and one component of the angular momentum as conserved quantities.

- Every one dimensional system with a Hamiltonian: Here, the energy is always a conserved quantity.

A typical counter-example is the double pendulum. It has two degrees of freedom, but the only conserved quantity is the energy. Such a system is called chaotic. Looking at the examples, one can recognize the advantage of integrable systems: Those systems can be analytically solved quite nicely, and therefore, are prime text book examples.

To study integrability in more complex systems, it turns out that a useful notion is the concept of Lax pairs. A Lax pair is a pair of squared matrices (in other words, they are operators) which takes values in the phase space of the problem setting. Usually, they are denoted by L and M, and depend on p and q in classical physics[1]. The defining property is the equation

$$\frac{\mathrm{d}}{\mathrm{dt}} L = [L, M] \,, \tag{6.1}$$

which the matrix pair has to fulfill if and only if the equations of motions are fulfilled. It is crucial to note that these matrices are not uniquely defined. For example, one can always add an identity or scale the matrix L by an arbitrary factor (unequal to zero).

Additionally, one can introduce a spectral parameter λ on which the matrices depend on. This factor is important in field theory to compute conserved quantities, and can always be introduced by

$$\tilde{L} = L + \lambda \, \mathbb{1} \,. \tag{6.2}$$

This spectral parameter is used to construct an infinite tower of conserved quantities

$$F_k(\lambda) = \operatorname{tr} L^k(\lambda) \,, \tag{6.3}$$

where we can expand the result as a power series in λ. As we mentioned above, we have to ensure, that those conserved quantities are actually in involution, so that $\{F_i, F_j\} = 0$ is true. For this, we have to find the **classical r-matrices** of the system. They are defined as

$$\{L_1, L_2\} = [r_{12}, L_1] - [r_{21}, L_2] \,, \tag{6.4}$$

where $L_1 := L \otimes 1$, $L_2 := 1 \otimes L$ and $r_{21} = P(r_{12})$ with $P(\cdot)$ being the permutation operator between the two spaces. It can be shown that all conserved quantities defined by (6.3) commute when classical r-matrices exist.

An example is the harmonic oscillator. It has the equations of motion

$$\dot{q} = p \,, \qquad \dot{p} = -\omega^2 q \,. \tag{6.5}$$

For these equations of motion, two different families of Lax pairs can be defined by

$$\begin{aligned} L_1 &= a \begin{pmatrix} p & b\,\omega\,q \\ \frac{\omega}{b} q & -p \end{pmatrix} \,, \qquad & M_1 &= \begin{pmatrix} 0 & \frac{b}{2}\omega \\ -\frac{1}{2\,b}\omega & 0 \end{pmatrix} \,, \\ L_2 &= a \begin{pmatrix} q & \frac{1}{b\,\omega} p \\ \frac{b}{w} p & -q \end{pmatrix} \,, \qquad & M_2 &= \begin{pmatrix} 0 & -\frac{1}{2\,b}\omega \\ \frac{2}{b}\omega & 0 \end{pmatrix} \,, \end{aligned} \tag{6.6}$$

---

[1] Note that it is also allowed that those matrices L and M depend on derivatives such as ∂_p or ∂_q. In this thesis, we are only interested in solutions depending only on coordinates itself, and not on derivatives.

where $a, b \in \mathbb{R}/\{0\}$. The solution known from the literature is L_1, M_1 with $a = b = 1$. A quick check shows that this exactly fulfills (6.1):

$$\dot{L}_1 = \begin{pmatrix} \dot{p} & \omega\, \dot{q} \\ \omega\dot{q} & -\dot{p} \end{pmatrix} = \begin{pmatrix} -\omega^2 q & \omega\, p \\ \omega p & \omega^2 q \end{pmatrix} = [L_1, M_1] \ . \tag{6.7}$$

Then, the classical R-matrix becomes

$$r_{12} = \frac{1}{q} \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \otimes \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} - \frac{1}{q} \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} . \tag{6.8}$$

When we introduce the spectral parameter with $\tilde{L} = L + \lambda \mathbb{I}$, the conserved quantities are:

$$\begin{aligned} F_1 &= 2\,\lambda\,, \\ F_2 &= 2\,\lambda^2 \ + \ 4\,H\,, \\ F_3 &= 2\,\lambda^3 \ + 12\,\lambda\,H\,, \\ F_4 &= 2\,\lambda^4 \ + 24\,\lambda\,H \ + \ 4\,H^2\,. \end{aligned} \tag{6.9}$$

As result, we find that the system has only one independent conserved quantity, the Hamiltonian.

As next step we extend this idea from classical mechanics to field theories. In those theories, an infinite amount of conserved quantities might be present. Therefore, we extend the concept of Lax pairs to the concept of a local **Lax connection** $A(\lambda)$ which is a matrix valued one-form. In $1+1$-dimensions, this one-form can be written as:

$$A(\lambda) = A_x(\lambda)\, dx + A_t(\lambda)\, dt\,. \tag{6.10}$$

This can be viewed as a connection and the Lax pair equation (6.1) becomes the flatness condition $dA = A \wedge A$, e.g.

$$\dot{A}_x(\lambda) - A_t^{'}(\lambda) + [A_x(\lambda), A_t(\lambda)] \ , \tag{6.11}$$

which has to be fulfilled if and only if the equations of motion are fulfilled. Here, we can reintroduce the spectral parameter again (if it is not present yet) by the formula:

$$\begin{aligned} \tilde{A}_x &= A_x + \lambda\ \mathbb{1}\,, \\ \tilde{A}_t &= A_t + \lambda\ \mathbb{1}\,. \end{aligned} \tag{6.12}$$

To find the Lax pair L and M from the Lax connection, we can use the definitions

$$L(\lambda) = \vec{P} \exp \int_0^R dx A_x(\lambda) \ , \qquad M(\lambda) = \left. A_t(\lambda) \right|_{x=0} \ , \tag{6.13}$$

where we defined the Lax pair for a compact space with length R, $\vec{P}$ being the path ordering operator. The conserved quantities can be found using the expansion of (6.3). For completeness, we also have to show that the conserved quantities are in involution. The defining condition here is

$$\{L_1(\lambda_1), L_2(\lambda_2)\} = [r_{12}(\lambda_1, \lambda_2), L_1(\lambda_1) \otimes L_2(\lambda_2)] \ . \tag{6.14}$$

6.2 Integrability Structures from Optimization

In this section, we are going to describe how to find the Lax pair, respectively the Lax connection, for different physical system as an optimization problem using the equations of motion. As we will see, we are going to need some additional input about the symmetry structure of our problem. To find such matrices, we are going to describe the search as an optimization problem and solve it with an suitable ansatz for L, M and the r-matrices. The framework is the same for the Lax pair and the Lax connection:

1. We start by sampling data points which fulfill the equations of motion. Note, that we do not need an analytical solution we can directly compute the time derivatives with the equations of motion: For the harmonic oscillator, we can sample q and p from a distribution, i.e. $p, q \sim \mathcal{N}(0,1)$, and then use the equations of motion to compute $\dot{q}$ and $\dot{p}$. This procedure has the advantage over using analytical formula that we do not have any unwanted correlations between coordinates. In our experiments, we use $\sim 10^5 \ - \ 10^6$ data points which can be easily extended to sampling new samples for every epoch.

2. As next step, we choose an ansatz for the Lax pair L, M. We usually start with the idea of having polynomials up to first order, and if the network does not converge we choose higher order polynomials. Another good guidance is the degree of the equations of motion. In general, we can use an arbitrary neural network which corresponds to approximating non-polynomial functions. Due to the fact that we are looking at standard examples of integrability in this proof of concept study and we are interested in finding analytical expressions, we decided to focus on simple ansätze. Given an ansatz we then optimize our network subject to the integrability loss described below.

3. Finally, we check the analytic results, whether they are actually equivalent to the differential equations. At this stage we can compute the conserved quantities as powers of tr L^k.

4. Additionally, we can find a solution for the r-matrices using the methods of 6.2.3. This can also be done in parallel to the search for the Lax pair, if the challenging part is to find a Lax pair where the conserved quantities are in involution.

As next step, we have to design our loss function. It has to ensure that the neural network is forced to find solutions to the problem in such a way that the following two conditions holds:

1. The equation

   $$\frac{\mathrm{d}}{\mathrm{dt}} L(\lambda) - [L(\lambda), M(\lambda)] = 0 \qquad \text{or} \qquad \dot{A}_x(\lambda) - A_t'(\lambda) + [A_x(\lambda), A_t(\lambda)] = 0 \tag{6.15}$$

   must hold for all sampled data points.

2. The system of equation, given by (6.15), must be equivalent to the defining equation of motions of the physical system.

To compute the derivatives, we can use the chain rules, and use the fact that, we control the dataset and manipulate it in any way, so for example we can evaluate the time derivative of L as follows

$$\frac{\mathrm{d}}{\mathrm{dt}}L = \frac{\partial L}{\partial p}\dot{p} + \frac{\partial L}{\partial q}\dot{q}\,, \tag{6.16}$$

where derivatives $\frac{\partial L}{\partial p}$ and $\frac{\partial L}{\partial q}$ are evaluated using auto-differentiation. The values $\dot{q}$ and $\dot{p}$ are part of the dataset.

6.2.1 Lax Pair

Let us start defining the loss functions. One central element is the Lax pair condition (6.1). To formulate that equation has a loss, we take the difference between left and right side, and take the absolute value of the components. Our choice here is mean-squared error, but it has the disadvantage of resulting in high loss numbers. Therefore, an other suitable choice would be the Huber loss function. We use

$$\mathcal{L}_{\mathrm{Lax}} = \left|\left|\dot{L} - [L, M]\right|\right|^2 \,, \tag{6.17}$$

where this norm is applied to each matrix component.
To fulfill the second part, we use a suitable ansatz: All equations of motion can be rewritten as

$$\dot{x}_i = f_i\left(x_i, \partial x_i, ...\right)\,, \tag{6.18}$$

where x_i is any quantity with a time derivative. Knowing that any time derivation of a variable originates from $\frac{dL}{dt}$, we can assume three things:

1. L must be linear in the variables x_k, e.g. $L = A_k x_k + B$.
2. The individual components of $\dot{L}$ have to be proportional to one of the $\dot{x}_k$ or has to vanish.
3. Each of the $\dot{x}_k$ must be proportional to at least one of the components of $\dot{L}$.

The first point defines the structure of our network making it only necessary to fit linear functions for L, the latter two points give us the structure of the loss function, leading to

$$\mathcal{L}_{\mathrm{L}} = \sum_{i,j} \min_k \left(||c_{ijk}\dot{L} - \dot{x}_k||^2, ||\dot{L}_{ij}||^2\right) + \sum_k \min_{i,j}\left(||c_{ijk}\dot{L}_{ij} - \dot{x}_k||^2\right)\,, \tag{6.19}$$

where $c_{ijk} = \frac{\sum_{\mathrm{batch}} \dot{L}_{ij}}{\sum_{\mathrm{batch}} \dot{x}_k}$ are the constants of proportionality. We use the same procedure for the right side of the equation, so for $[L, M]$:

$$\mathcal{L}_{\mathrm{LM}} = \sum_{i,j} \min_k \left(||\tilde{c}_{ijk}\left[L, M\right]_{ij} - f_k||^2, ||\left[L, M\right]_{ij}||^2\right) + \sum_k \min_{i,j}\left(||\tilde{c}_{ijk}\left[L, M\right]_{ij} - f_k||^2\right)\,, \tag{6.20}$$

where $\tilde{c}_{ijk} = \frac{\sum_{\text{batch}}[L,M]_{ij}}{\sum_{\text{batch}} f_k}$. Note the redundancies in the loss term $\mathcal{L}_{\text{LM}}$, a combination of $\mathcal{L}_{\text{Lax}}$ and $\mathcal{L}_{\text{L}}$ would be sufficient, but the additional term facilitates training.

The last term of the loss exists to prevent mode collapse. Our solution here is to set a lower limit (in this example 1) on the sum of absolute values of the all components of A_k:

$$\mathcal{L}_{\text{MC}} = \max\left(1 - \sum |A_{ij}|\,, 0\right)\,. \tag{6.21}$$

The exact structure of the loss $\mathcal{L}_{\text{MC}}$ can be varied and we can use this to favor specific solutions of the neural network. The total loss is

$$\mathcal{L}_{\text{Lax-Pair}} = \alpha_1 \mathcal{L}_{\text{Lax}} + \alpha_2 \mathcal{L}_{\text{L}} + \alpha_3 \mathcal{L}_{\text{LM}} + \alpha_4 \mathcal{L}_{\text{MC}}\,, \tag{6.22}$$

where in our experiments we set $\alpha_1 = \alpha_2 = 1$, $\alpha_4 = 10$ and α_3 to 1 or 0.

6.2.2 Lax connection

The same approach can be used for the problem of finding Lax connections: We have to replace every L with an A_x, every M with an A_t. Additionally, we have to take the $\partial_x A_t$ into account. This results in the loss components

$$\begin{aligned}
\mathcal{L}_{\text{Lax}} =& \sum_{ij} \left|\left|\dot{A}_x - A_t' + [A_x, A_t]\right|\right|^2 \\
\mathcal{L}_{A_x} =& \sum_{i,j} \min_k \left(||c_{ij}\dot{A}_{x,ij} - \dot{x}_k||^2, ||\dot{A}_{x,ij}||^2\right) + \sum_k \min_{i,j} \left(||c_{ijk}\dot{A}_{x,ij} - \dot{x}_k||^2\right) \\
\mathcal{L}_{A_t} =& \sum_{i,j} \min_k \left(||c_{ijk}\left[-A_t' + [A_x, A_t]\right]_{ij} - f_k||^2, ||\left[-A_t' + [A_x, A_t]\right]_{ij}||^2,\right) \\
&+ \sum_k \min_{i,j} \left(||c_{ijk}\left[-A_t' + [A_x, A_t]\right]_{ij} - f_k||^2\right) \\
\mathcal{L}_{\text{MC}} =& \max\left(1 - \sum |A_k|\,, 0\right)\,.
\end{aligned} \tag{6.23}$$

The total loss for the Lax connection is

$$\mathcal{L}_{\text{Lax-Connection}} = \alpha_1 \mathcal{L}_{\text{Lax}} + \alpha_2 \mathcal{L}_{A_x} + \alpha_3 \mathcal{L}_{A_t} + \alpha_4 \mathcal{L}_{\text{MC}}\,. \tag{6.24}$$

For complex-valued matrices, we interpret the complex space as $\mathbb{R}^2$, and therefore, use it as an additional dimension for the summations and the minima.

6.2.3 Classical R-matrices

We use a similar approach for the r-matrices as for the Lax pair. For a given Lax pair, we can learn the r-matrices directly using

$$\mathcal{L}_{\text{R}} = ||\{L_1, L_2\} - [r_{12}, L_1] + [r_{21}, L_2]\,||^2\,. \tag{6.25}$$

6.2.4 Linear combinations within the equations of motion

In the descriptions above, we assumed that linear combinations of the equations of motion do not occur. For most systems, it is possible, to find a suitable Lax pair, but not for all of them. One way of extending the frame work is to focus on the loss term $\mathcal{L}_{\mathrm{L}}$, while ignoring $\mathcal{L}_{\mathrm{LM}}$ (setting $\alpha_3 = 0$. We compute the loss term $\mathcal{L}_{\mathrm{L}}$ for each equation of motion individually, so when having several equations of motion:

$$\begin{aligned} \dot{x}_1 &= f_1\left(x_1, x_2\right) \, , \\ \dot{x}_2 &= f_2\left(x_1, x_2\right) \, , \\ \dot{x}_3 &= \dots \, , \end{aligned} \tag{6.26}$$

we first set all $\dot{x}_i = 0$, except for $\dot{x}_1$, compute the loss term $\mathcal{L}_{\mathrm{L}}$, and then move on to the next equations of motion etc. This method avoids linear combinations in the equations of motion, and therefore, the presented framework still works. Even though this workaround is not suitable for $\mathcal{L}_{\mathrm{LM}}$, the right hand side still results in the correct linear combination of the equations of motion due to the loss component $\mathcal{L}_{\mathrm{Lax}}$.

6.3 Experiments

After presenting the optimization algorithm, we discuss several examples of the method. We start with the harmonic oscillator as an example of classical mechanics. Then, we continue with the Korteweg–de Vries equation (KdV) (which model shallow water waves) as first field theoretical example, and continue with two important examples from theoretical physics: the Heisenberg model and the principal chiral model.

6.3.1 Architecture of the Networks

We start with a short description of suitable neural networks. In general, we can use any neural network architecture and train the weight using gradient-based optimization of the neural network. There are two general ideas, we can use to build a neural network:

1. We can use standard multi-layer perceptrons to model the matrices. The advantage is that it should be straightforward to train, their training behavior is well understood and therefore, we should be able to learn the functions. The problem is, that we would learn a general function, we might be able to learn new Lax pairs, but the functions used by the neural network is too complicated to fit. Additionally, we are using for the matrix L the ansatz $L = A_k x_k + B$. To model such a function, we would have to use a neural network with zero hidden layer, which contradicts the advantages described above.

2. We can fit polynomials as solutions, using every prefactor as a trainable weight. It turned out that such approaches only have a rather small number of weights. Due

to the complicated constrains the solution has to fulfill, the neural network depends heavily on the initialization.

To avoid those limitations, we use a combination of the two approaches which gives us on the one hand side enough weights to be less initialization dependent, but also gives us directly analytic formulae. For each prefactor a of the polynomial, we fit a vector $\vec{a}$ with dimension $2\ n + 1$. Then, the prefactor a is calculated using a softmax on the weights, multiplied with a vector of fixed numbers, i.e.

$$a = \text{softmax}\,(\vec{a}) \cdot \vec{v} \qquad \text{with} \qquad \vec{v} = \begin{pmatrix} n \\ \vdots \\ -n \end{pmatrix} . \tag{6.27}$$

Using such an ansatz, the neural network is able to change the value of the prefactors much faster compared to only one trainable weight. It is also important to note that for the matrix M we should always start with the lowest possible degree in the polynomial to support the neural network in the training process. Note, that the training process is still depending on the initialization, and therefore, one should start the same neural network using a different initialization. To support the convergence of the neural network and to avoid finding linear combinations of the equations of motion, we can use the values $\vec{a}$ to redefine the prefactors within L. Assuming, we have multiple $x_1, \ldots, x_n$, we can define the prefactor a_k as

$$a_k = \sigma \left(\sum_i a_{k,i} - \sum_{j \neq k} \sum_i a_{j,i} \right) \ \text{softmax}\,(\vec{a_k}) \cdot \vec{v} \,, \tag{6.28}$$

with σ being the sigmoid function. This sigmoid function has the effect that the neural network automatically singles out one x_k. Note, that this is one way of improving the performance of the neural network. For many systems, it is more suitable to check the structure of the equations of motion and look for the symmetry group of them. In many cases (for example the Heisenberg model), we have an underlying SU(2)-symmetry, and therefore, it is expected that the Lax pair contains factors of Pauli matrices within the neural network. This speeds up training due to the fact that the structure of L is already constrained. For the training, we use an Adam optimizer with a learning rate of 0.01 if not otherwise stated. We trained our parameters for 50 000 steps with a batch size of 2000. In each step, we trained the network for L and M separately.

6.3.2 Harmonic Oscillator

We start with the example of the harmonic oscillator. The Hamiltonian is

$$H = \frac{1}{2}\, p^2 + \frac{\omega^2}{2}\, q^2 \,, \tag{6.29}$$

where we used $\omega = 2$. In general, we can use an arbitrary ω, but this would lead to a more complicated structure of the problem due to higher degrees of the polynomials. The equations of motion for this problem are

$$\dot{q} = p \,, \qquad \dot{p} = -\omega^2 q \,, \tag{6.30}$$

which we used to sample 10^5 data points with $p, q \sim \mathcal{N}(0,1)$. As ansatz for L and M, we use

$$\begin{aligned} L_{ij}(p,q) &= a_{ij} + b_{ij} q + c_{ij} p \,, \\ M_{ij}(p,q) &= d_{ij} + e_{ij} q + f_{ij} p \,, \end{aligned} \tag{6.31}$$

with $i, j = \{1, 2\}$ (so for 2×2-matrices). We restrict the sum over the absolute value of b_{ij} and c_{ij} to avoid mode collapse

$$\sum_{i,j=1}^{2} |b_{ij}| \geq \frac{1}{2} \qquad \text{and} \qquad \sum_{i,j=1}^{2} |c_{ij}| \geq \frac{1}{2} \,, \tag{6.32}$$

using the loss component $\mathcal{L}_{\mathrm{MC}}$. We find with this ansatz the Lax pair given by

$$L = \begin{pmatrix} 0.437\, q & -0.073\, p \\ -0.666\, p & -0.437\, q \end{pmatrix} , \qquad M = \begin{pmatrix} 0.001 & 0.329 \\ -3.043 & -0.001 \end{pmatrix} , \tag{6.33}$$

where we do not allow linear combinations of the differential equations. A check shows that both sides of the Lax pair condition (6.1) match accurately

$$\frac{dL}{dt} = \begin{pmatrix} 0.437\, \dot{q} & -0.073\, \dot{p} \\ -0.666\, \dot{p} & -0.437\, \dot{q} \end{pmatrix} = \begin{pmatrix} 0.441\, p & 0.288\, q \\ 2.660\, q & -0.441\, p \end{pmatrix} = [L, M] \tag{6.34}$$

We check that $\mathrm{tr} L^2 \sim H$:

$$L^2 = \begin{pmatrix} 0.048618 p^2 + 0.190969 q^2 & 0 \\ 0 & 0.048618 p^2 + 0.190969 q^2 \end{pmatrix} \Rightarrow \mathrm{tr} L^2 \approx 0.1\ H \,. \tag{6.35}$$

Therefore, we can see that our method is perfectly able to reproduce the Lax pair in the literature.

We train also for the corresponding R-matrices and find as a result

$$r_{12} = \begin{pmatrix} 0 & 0.92 - \frac{0.42}{p} \\ -1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 0 & 0.92 \\ -1 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 0.92 \\ -1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 0 & 0.92 - \frac{0.42}{p} \\ -1 & 0 \end{pmatrix} , \tag{6.36}$$

which solves perfectly equation (6.14).

We also want to mention, that this training process is heavily instable and we used several runs with different initialization to find this solution. Most of the training processes where not converging at all. Comparing the harmonic oscillator to the other examples, it turns out that this is one of the more complicated examples due to the independence of p and q, while in the other examples, we were able to restrict L using a symmetry analysis of the example. We suggest a work around for this problem: one possibility here is to simply use a different approach using only a constant for M. This corresponds to the most simple ansatz for the Lax pair but leads to a bit more stable convergence (and is suitable in this case). Additionally, we also test a standard multi-layer perceptron as neural network for M with two hidden layers with 200 neurons and a tanh-activation, and a 2×2-dimensional output layer. In this case, the training process converges in every case. Here, we are also able to fit an analytical solution to M due to the simple structure of the analytic solution.

6.3.3 Korteweg-de Vries equation

We examine the Korteweg–de Vries equation as second system. It is used to describe waves on shallow water surfaces. The equation of motion is given by

$$\dot{\phi} + \phi^{'''} - 6\,\phi\,\phi^{'} = 0\,. \tag{6.37}$$

It is an one-dimensional realization of a classical field theory and a good first example to find Lax connections. For the sampling process, we sample $\phi\,,\ \phi^{'}\,,\ \phi^{''}\,,\ \phi^{'''} \approx \mathcal{N}(0,2)$ and compute $\dot{\phi}$ using equation (6.37).

The ansatz we use is determined by the structure of the problem: We expect the matrix A_x to be linear in $\phi(x)$, while the matrix A_t should be a polynomial up to second order. Computing $\frac{dA_t}{dx}$ in the defining equation of the Lax connection, we expect that M depends only on the terms ϕ, $\phi^{'}$ and $\phi^{''}$, and not on $\phi^{'''}$. Using a 2×2-matrix as ansatz, we find

$$\begin{aligned} A_x &= \begin{pmatrix} 1.6\phi - 0.2 & -0.8 \\ 0.2 & 1.7\phi + 0.2 \end{pmatrix}, \\ A_t &= \begin{pmatrix} -4.9\phi^2 - 1.6\phi^{''} - 0.1 & 0.1\phi^2 - 0.3 \\ 0.1 & -5.0\phi^2 - 1.7\phi^{''} + 0.1 \end{pmatrix}, \end{aligned} \tag{6.38}$$

where we rounded entries to 10^{-1}. This result shows, that even a 1×1-matrix would be enough to model equation (6.37).

To force the neural network to use the off-diagonals as well, we change the loss term $\mathcal{L}_{\mathrm{MC}}$ in such a way that only off-diagonal elements enter this loss term. As a result, we find:

$$\begin{aligned} A_x &= \begin{pmatrix} -1.7\phi & 1.7\phi + 1.0 \\ 1.7\phi + 1.0 & -1.7\phi \end{pmatrix}, \\ A_t &= \begin{pmatrix} 5.0\phi^2 + 1.7\phi^{''} & -5.0\phi^2 - 1.7\phi^{''} - 0.5 \\ -5.0\phi^2 - 1.7\phi^{''} - 0.5 & 5.0\phi^2 + 1.7\phi^{''} \end{pmatrix}, \end{aligned} \tag{6.39}$$

which is now much closer to the literature and again provides a pair which fulfills the required conditions, i.e.

$$\begin{aligned} &\frac{\partial A_x}{\partial t} - \frac{\partial A_t}{\partial x} + [A_x, A_t] = \\ &= \begin{pmatrix} -1.7\dot{\phi} & 1.7\dot{\phi} \\ 1.7\dot{\phi} & -1.7\dot{\phi} \end{pmatrix} - \begin{pmatrix} 10.0\phi^{'}\phi + 1.7\phi^{'''} & -10.0\phi^{'}\phi - 1.7\phi^{'''} \\ -10.0\phi^{'}\phi - 1.7\phi^{'''} & 10.0\phi^{'}\phi + 1.7\phi^{'''} \end{pmatrix} + \mathcal{O}(0.1) \approx 0 \end{aligned} \tag{6.40}$$

6.3.4 Heisenberg Magnet

Let us start with the Heisenberg magnet for a ferromagnet with an SO(3)-symmetry group, one of the important field theoretical examples. The Hamiltonian of the Heisenberg magnet is

$$H = \frac{1}{2} \int dx \; \vec{S}^2 \left(x \right) , \tag{6.41}$$

where $\vec{S}^2 = 1$ and the components satisfy as an additional constraint

$$\left\{ S_a \left(x \right) , S_b \left(y \right) \right\} = \epsilon_{abc} S_c \left(x \right) \delta \left(x - y \right) \; . \tag{6.42}$$

The equations of motion read

$$\dot{\vec{S}} = \{ H, \vec{S} \} = -\vec{S}(x) \times \vec{S}^{''}(x) \, . \tag{6.43}$$

As we can see, we have an SO(3) ≈ SU(2) invariant structure in the equations, we can utilize to find the Lax connection. Assuming that the Lax connection must be invariant under an SU(2) transformation on the $\vec{S}$ we use as ansatz for A_x:

$$A_x = a \; \vec{\sigma} \vec{S} + B \, , \qquad \text{with} \quad a \in \mathbb{C} \, , B \in \mathbb{C}^{(2 \times 2)} \, . \tag{6.44}$$

For the matrix A_t, we use a polynomial ansatz up to second order in S_i and $S_i^{'}$, the second derivative should be irrelevant, it would turn to a third order derivative in Lax connection equation (6.11). Therefore, the ansatz is

$$A_t = C \; S_i + D \; S_i^{'} + E \; S_i \; S_j + F \; S_i \; S_j^{'} + G \; S_i^{'} \; S_j^{'} \, , \qquad \text{with} \quad C \, , \; D \, , \; E \, , \; F \, , \; G \; \in \mathbb{C}^{(2 \times 2)} \, . \tag{6.45}$$

Using this ansatz (which looks reasonable) we find the following formulae for the Lax connection A_x and A_t:

$$\begin{aligned}
A_x =& - \; \mathrm{i} \; \vec{\sigma} \vec{S} + 0.3 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\
A_t =& \begin{pmatrix} 2 \; \mathrm{i} \; S_z & 2 \; \mathrm{i} \; S_x + 2 S_y \\ 2 \; \mathrm{i} \; S_x - 2 S_y & - \; \mathrm{i} \; S_z \end{pmatrix} \\
& + \begin{pmatrix} \mathrm{i} \; S_y^{'} S_x - \; \mathrm{i} \; S_x^{'} S_y & -S_z^{'} S_x + S_x^{'} S_z + \; \mathrm{i} \; (S_z^{'} S_y - S_y^{'} S_x) \\ + S_z^{'} S_x - S_x^{'} S_z + \; \mathrm{i} \; (S_z^{'} S_y - S_y^{'} S_x) & - \; \mathrm{i} \; S_y^{'} S_x + \; \mathrm{i} \; S_x^{'} S_y \end{pmatrix} \\
=& 2 \; \mathrm{i} \; \vec{\sigma} \vec{S} + \; \mathrm{i} \; \epsilon_{ijk} \sigma_i S_j S_k^{'} \; ,
\end{aligned} \tag{6.46}$$

where we rounded at the least shown digit. Comparing this with the literature, we find exactly the same results as in [57].

Here, we can see the advantages, as well as the disadvantage of the method: The neural network is perfectly able to find the correct structure within A_t, but we also have to use our physical intuition on A_x. Starting the search without this restriction, we have not been able to find a solution.

6.3.5 Non-linear sigma model

The next important class of integrable field theory models is given by non-linear sigma models in $1+1$ dimensions. We are interested in $\mathrm{O}(N)$ sigma models with fields in S^{N-1}. The Lagrangian of these models is:

$$\mathcal{L} = -\mathrm{Tr}\ \left(J_\mu J^\mu\right), \qquad J_\mu = (\partial_\mu g) g^{-1}\ , \qquad \mu = 0,1 \tag{6.47}$$

This system obeys the following equations:

$$\begin{aligned} \partial_\mu J^\mu &= 0\ , \\ \partial_\mu J_\nu - \partial_\nu J_\mu - [J_\mu, J_\nu] &= 0\ , \end{aligned} \tag{6.48}$$

with $\mu = 0,1$, corresponding to the spatial and time like directions.

In the following, we study the two special cases $N = 3$ and $N =$. The case $N = 3$ is related to the Sine-Gordon model which can be seen by appropriately re-writing the equations of motion (cf. Chapter 6.3 of [58] and [59]), whereas the case $N = 4$ is related to the principal chiral model.

Sine-Gordon equation

The Sine-Gordon equation is given by

$$\frac{\partial^2}{\partial x \partial t}\phi - \sin(\phi) = 0\,. \tag{6.49}$$

For this equation, we use a known solution of the Sine-Gordon equation to sample data points

$$\phi(x,t) = 4\ \arctan\left(\exp\left(\gamma\left(x + t - v\ (x-t)\right) + \delta\right)\right)\ , \qquad \text{with}\, \gamma^2 = \frac{1}{1-v^2}\,, \tag{6.50}$$

where we sample $v \sim \mathcal{U}(-0.9, 0.9)$ and $\delta \sim \mathcal{N}(1,1)$.

As ansatz for the Lax connection, we use a similar approach as before. For the matrix A_x, we are able to see that it should only depend on $\frac{\partial \phi}{\partial x}$:

$$A_{x,ij} = a_{ij}\phi^{'} + b_{ij}\ , \tag{6.51}$$

while for the matrix A_t, we expect a dependence on either sin or cosine:

$$A_{t,ij} = c_{ij}\phi + d_{ij}\frac{\partial \phi}{\partial x} + e_{ij}\cos\phi + f_{ij}\sin\phi\,. \tag{6.52}$$

Using complex coefficients, we find for this ansatz

$$\begin{aligned} A_x &= \begin{pmatrix} 0.5\phi^{'} - 0.7 - 0.6\ \mathrm{i} & 0.9\phi^{'} + 0.2\ \mathrm{i}\ \phi^{'} \\ -0.5\phi^{'} + 0.1\ \mathrm{i}\ \phi^{'} + 0.8 + 0.4\ \mathrm{i} & -0.5\phi^{'} + 0.7 + 0.6\ \mathrm{i} \end{pmatrix}\ , \\ A_t &= \begin{pmatrix} -0.2\cos\phi + 0.2\ \mathrm{i}\ \cos\phi + 0.2\sin\phi - 0.2\ \mathrm{i}\ \sin\phi & 0.5\sin\phi - 0.2\ \mathrm{i}\ \sin\phi \\ 0.2\cos\phi - 0.2\ \mathrm{i}\ \cos\phi & 0.2\cos\phi - 0.2\ \mathrm{i}\ \cos\phi - 0.2\sin\phi + 0.2\ \mathrm{i}\ \sin\phi \end{pmatrix}\,. \end{aligned} \tag{6.53}$$

The Lax connection is satisfied for these matrices

$$
\begin{aligned}
\frac{dA_x}{dt} &= \begin{pmatrix} 0.5\frac{d^2\phi}{dxdt} & 0.9\frac{d^2\phi}{dxdt} + 0.2\ \mathrm{i}\ \frac{d^2\phi}{dxdt} \\ -0.5\frac{d^2\phi}{dxdt} + 0.1\ \mathrm{i}\ \frac{d^2\phi}{dxdt} & -0.5\frac{d^2\phi}{dxdt}\ \mathrm{i} \end{pmatrix} = \\
&= \begin{pmatrix} 0.5\sin\phi & 0.9\sin\phi + 0.2\ \mathrm{i}\ \sin\phi \\ -0.5\sin\phi + 0.1\ \mathrm{i}\ \sin\phi & -0.5\sin\phi \end{pmatrix} = \frac{dA_t}{dx} - [A_x, A_t]\ .
\end{aligned}
\tag{6.54}
$$

As we can see, the equation is only fulfilled iff the Sine-Gordon equation is fulfilled as well.

Principal chiral model

For the case $N = 4$, the equations of motion are given by (6.48), which corresponds to an SO(4) symmetry. Typically, this system is examined using the split $\mathrm{SO}(4) \approx \mathrm{SU}(2)\times\mathrm{SU}(2)$, which we are using as well. Then, we can use the SU(2) symmetry, and use an ansatz which is proportional to the Pauli-matrices:

$$
A_x = a\ \vec{\sigma}\vec{J}_x + b\ \vec{\sigma}\vec{J}_t\,, \qquad \text{with} \quad a, b \in \mathbb{C}\,, \tag{6.55}
$$

whereas the matrix A_t a polynomial of second order, and is otherwise unconstrained. Note, that with this ansatz it is expected that we are going to find linear equations between the different equations of motion and we therefore, have to change the loss function as described in 6.2.4. We sample all variables using $\mathcal{N}(0, 2)$ for the sampling and the equations of motion (cf. Equation (6.48)) for 10^6 samples. As solutions we find for the Lax connection:

$$
\begin{aligned}
A_x &= (0.295 +\ \mathrm{i}\ 0.205)\,\vec{\sigma}\vec{J}_x + (-0.512 +\ \mathrm{i}\ 0.165)\,\vec{\sigma}\vec{J}_t\,, \\
A_t &= (0.515 -\ \mathrm{i}\ 0.162)\,\vec{\sigma}\vec{J}_x + (-0.283 -\ \mathrm{i}\ 0.193)\,\vec{\sigma}\vec{J}_t\,,
\end{aligned}
\tag{6.56}
$$

where the matrix A_t has perfectly the shape of the Pauli-matrices. We check that our solution is equivalent to the equations of motion

$$
\begin{aligned}
&\dot{A}_x - A_t' + [A_x, A_t] = \\
&= (0.295 +\ \mathrm{i}\ 0.205)\,\vec{\sigma}\dot{\vec{J}}_x + (-0.512 +\ \mathrm{i}\ 0.165)\,\vec{\sigma}\dot{\vec{J}}_t \\
&\quad + (0.515 -\ \mathrm{i}\ 0.162)\,\vec{\sigma}\vec{J}_x{}' - (0.283 +\ \mathrm{i}\ 0.193)\,\vec{\sigma}\vec{J}_t{}' \\
&\quad - (0.295 +\ \mathrm{i}\ 0.205)\,(0.283 +\ \mathrm{i}\ 0.193)\ \ 2\ \epsilon^{abc} J_t^b J_x^c \\
&\quad + (0.515 -\ \mathrm{i}\ 0.162)\,(-0.512 +\ \mathrm{i}\ 0.165)\ \ \mathrm{i}\ 2\ \epsilon^{abc} J_t^b J_x^c \\
&\approx (-0.512 +\ \mathrm{i}\ 0.165)\,\vec{\sigma}\left(\dot{\vec{J}}_t - \vec{J}_x{}'\right) \\
&\qquad\quad + (0.283 +\ \mathrm{i}\ 0.193)\,\vec{\sigma}\left(\dot{\vec{J}}_x - \vec{J}_t{}'\right) - 2\ \ (0.283 +\ \mathrm{i}\ 0.193)\ \ \epsilon^{abc}\sigma^a J_t^b J_x^c\ ,
\end{aligned}
\tag{6.57}
$$

The solution we found can be matched to the known solution presented in [60].

6.4 Integrable vs Non-Integrable Perturbations

An important field of research for integrable systems is the examination of perturbations of an integrable system. In this section, we focus on two integrable systems, the two dimensional harmonic oscillator and the Heisenberg model, and examine different types of perturbations. The magnitude of the perturbation here is controlled by the parameter ϵ. We start with the unperturbed system with a correctly initialized neural network and then increase the parameter ϵ. When comparing the trend of the training loss between the perturbed and the unperturbed system, we are able to see whether the neural network is able to compensate the perturbation and still find a true Lax pair. Here, we allow linear combinations between the different equations of motion.

6.4.1 Two-dimensional harmonic oscillator

We start with the harmonic oscillator in two dimensions[2]. Here, we are comparing a quadratic perturbation with a fourth order perturbation. The Hamiltonian is

$$\begin{aligned} H_1 =& \frac{1}{2}\left(p_x^2 + p_y^2 + x^2 + y^2 + \epsilon xy\right) \,, \\ H_2 =& \frac{1}{2}\left(p_x^2 + p_y^2 + x^2 + y^2 + \epsilon x^2 y^2\right) \,. \end{aligned} \tag{6.58}$$

In both Hamiltonians, the parameter ϵ scales the perturbation. The Hamiltonian H_1 is integrable because it describes a coupled harmonic oscillator, whereas the Hamiltonian H_2 is not integrable.

To initialize the neural network on a known solution, we use a 4×4-dimensional Lax pair with two separated systems in the upper-left and lower-right part of the matrices. We use the same structure as before for L (see eq. (6.27)), while modeling M as a neural network with two hidden layers with 200 neurons each and tanh-activation on the hidden layer, followed by an output layer.

We draw 100000 samples of x, y, p_x and p_y from $\mathcal{N}(0,1)$, and compute $\dot{x}$, $\dot{y}$, $\dot{p}_x$ and $\dot{p}_y$ using:

$$\begin{aligned} &\dot{x} = p_x \,, \qquad \dot{y} = p_y \,, \qquad \dot{p}_x = -x - \epsilon y \,, \qquad \dot{p}_y = -y - \epsilon x \,, \\ &\dot{x} = p_x \,, \qquad \dot{y} = p_y \,, \qquad \dot{p}_x = -x - \epsilon\, x\, y^2 \,, \qquad \dot{p}_y = -y - \epsilon\, y\, x^2 \,. \end{aligned} \tag{6.59}$$

Note that, for each value of ϵ we have to either recompute the time derivatives or sample a new dataset.

We start the training process with $\epsilon = 0$ and increased ϵ every 1000 steps for 0.05 until it reaches ϵ_{final}. We train the neural networks for $\epsilon_{\text{final}} = (0, 0.001, 0.01, 0.1, 0.5, 1)$ using the Adam optimizer with learning rate 10^{-3}. In total, we train each neural network for 80 000 steps. The evolution of the training loss, and therefore the violations of the Lax

---

[2] Note that the harmonic oscillator as an one dimensional system is integrable with any kind of perturbation because it always has one conserved quantity (the energy), and only one degree of freedom.

pair constraints, can be seen in Figure 6.1 at the top. For the integrable perturbation, the neural networks can adapt the values for L and M, and have similar losses as for the unperturbed system. In contrast to that, the neural networks are not able to adapt to the non-integrable perturbation and the training does no longer converge, i.e. hierarchically larger loss values are encountered compared to the unperturbed system. In the figure, we show the median of 10 runs for each ϵ and perturbation type, the curves are smoothed using a Gaussian filter with $\sigma = 10$.

Finally, we want to point out that such an increase of the loss is only a hint for a non-integrable perturbation. It might also be the consequence of a too simple ansatz for the Lax pair or due to a too high or too low learning rate.

6.4.2 Heisenberg Model

As second example for an integrable system with perturbations, we present the Heisenberg model as in section 6.3.4. Here, we compare two different quadratic perturbations: The integrable one is known from the Landau-Lifschitz equation ($S \wedge JS$) as

$$\dot{\vec{S}} = -\vec{S}(x) \times \vec{S}''(x) - \vec{S} \times J\vec{S} \qquad \text{with } J = \begin{pmatrix} -\epsilon & 0 & 0 \\ 0 & \epsilon & 0 \\ 0 & 0 & 0 \end{pmatrix} , \tag{6.60}$$

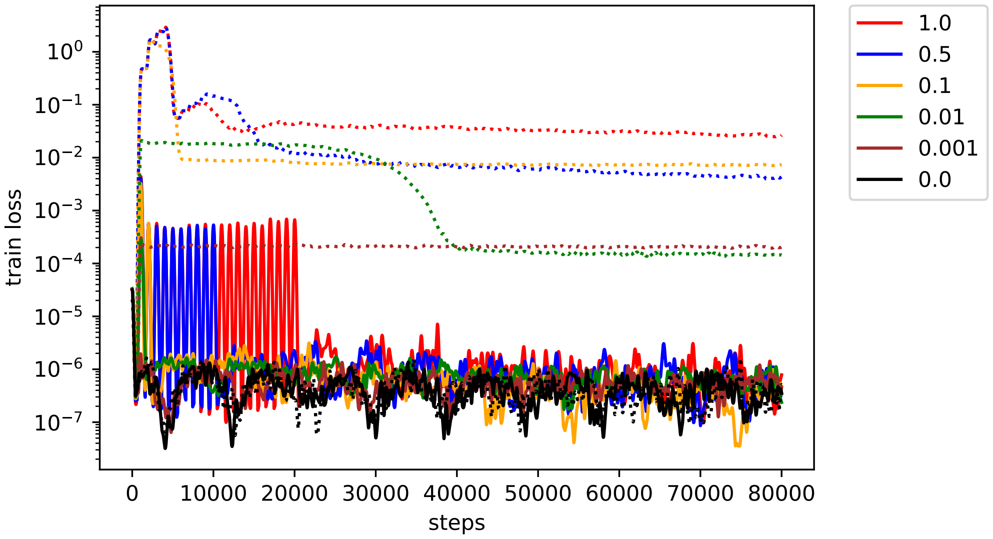
while the second system has a different quadratic perturbation, i.e.

$$\dot{\vec{S}} = -\vec{S}(x) \times \vec{S}''(x) + \begin{pmatrix} \epsilon\, S_x^2 \\ -\epsilon\, S_y^2 \\ 0 \end{pmatrix} . \tag{6.61}$$

The procedure is similar as for the Hamiltonian oscillator: We start with a correctly initialized neural network for the unperturbed Heisenberg model, and increase the parameter ϵ with small steps of 0.05 until we reach ϵ_{final}. Again, we use $\epsilon_{\text{final}} = (0, 0.001, 0.01, 0.1, 0.5, 1)$.

We use a 4 × 4-dimensional ansatz for the Lax connection, initializing the neural network with a correct solution in the upper-left part of the matrices. For A_x, we use a linear approach as before. A_t consists of two parts: one polynomial which is used for the initialization, whereas we add a multi-layer perceptron with two hidden layer, with 200 neurons each and tanh-activation on the hidden layers.

Again, we trained the neural network for 80 000 steps, with Adam optimizer with a learning rate of 0.001. The results are presented in the bottom of Figure 6.1. We can see, as for the harmonic oscillator, that the neural network is able to adapt the Lax connection to the integrable perturbation and converge to the same value as the non-perturbed Lax connection. For the non-integrable perturbation, this is not the case. Therefore, with our method we are able to distinguish between integrable and non-integrable systems.



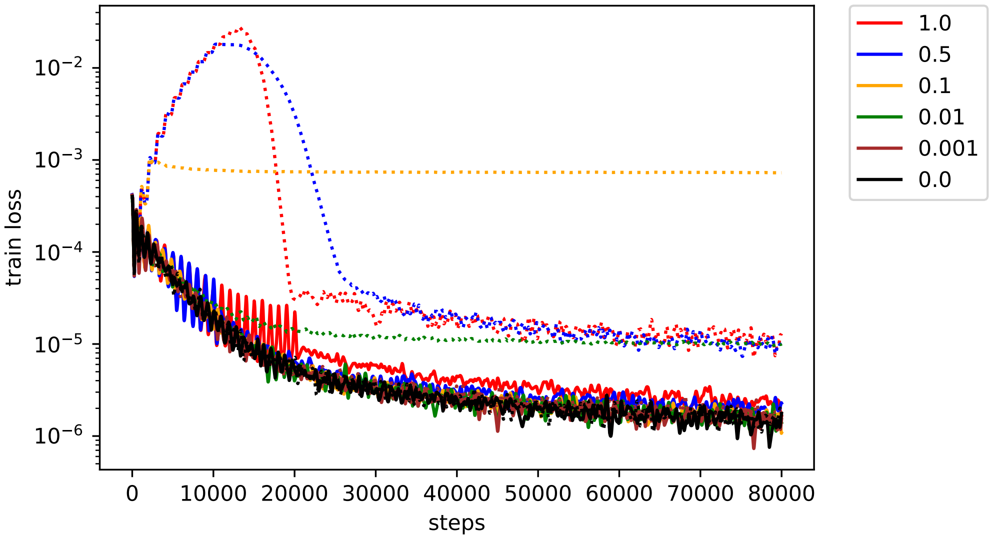


Figure 6.1: In these two plots we compare the different behavior of the training loss for integrable (solid lines) and non-integrable (dotted lines) perturbations for the two-dimensional harmonic oscillator and the Heisenberg model. In the **upper** graphic the results for two-dimensional harmonic oscillator is displayed, whereas in the **lower** one we can see the results for the Heisenberg model. In both cases, we start with no perturbation and it grows linearly with $\epsilon_{\text{step}} = 0.05$ every 1000 steps. The raise of the perturbation parameter ϵ can be seen in the beginning for large values of ϵ as steps in the graph. For integrable perturbations, the neural network is able to adapt the Lax pair to the perturbation, therefore, after some time, the training error corresponds to the unperturbed system (displayed in black). For non-integrable perturbations, the neural network is not able to compensate. The training error grows proportional to the perturbation (note the log scale of the y-axis). After many epochs, the neural network finds a different minima which still leaves the training error multiple orders of magnitude above the unperturbed system.

Part III

Effects of Symmetries

Chapter 7

Basics of Dualities and Exceptional Field Theory

After presenting in the last part different methods how to find structures, symmetries and conserved quantities in physical systems, we are going to focus in this part of the thesis on the effects of symmetries in string theory and their impact on the phase space of this theory. In the first chapter of the this part, we review T-duality and its mathematical implications to find the symmetry group, followed by the extension to the S- and U-duality. Afterwards, we introduce the general idea of exceptional field theory (based on the review [61]) and conclude with the construction of duality chains of T- and U-dualities. In the next two chapters, we focus on our recent research projects, published in [25] and [26]: In chapter 8, we use exceptional field theory to find previously unknown non-geometric fluxes and their consequences, whereas in chapter 9, we exploit the duality chains to find the commutativity behavior of open strings in non-geometric backgrounds.

7.1 Dualities

In the introduction 1.3, we already introduced the main concepts and notations of string theory. Here, we are now going to focus on the dualities, which connect the different string theories. In figure 1.1, we can see the connections by the T- and S-dualities. We show T-duality explicitly on the level of the mass formula to find the underlying symmetry group. In the next step, we focus on the symmetry group of S-duality and extend the concept to the symmetry groups of U-dualities. During all of these chapters we will set $\alpha' = 2$. We are going to follow the text book [19] where more details of the derivations are given.

7.1.1 T-Duality

We start this review on T-duality by looking at a one-dimensional compactification of a closed bosonic string with the world-sheet coordinates X^i. We start our example by compactifying the bosonic string over a circle with radius R in dimension 25. The coordinate for this circle is denoted by x. Then, we can identify the points accordingly as

$$x \sim x + 2\pi R\,, \quad L \in \mathbb{Z}\,. \tag{7.1}$$

This identification also has implications on the coordinate X^{25} leading to the constraint that

$$X^{25}(\sigma + 2\pi, \tau) = X^{25}(\sigma, \tau) + 2\pi R L\,. \tag{7.2}$$

In this equation L counts how often the string is winded around the compactified dimension and is called winding number. A string with winding number cannot be "unwinded" or shrinked to a point, and therefore, the winding number is a conserved quantity of each string state.

Using condition (7.2), we are able to find the mode expansion for the string in the compactified dimension,

$$\begin{aligned} X^{25}(\sigma,\tau) =& x^{25} + \alpha' p^{25}\tau + LR\sigma \\ &+ i\sqrt{\frac{\alpha'}{2}} \sum_{n\neq 0} \frac{1}{n}\left(\alpha_n^{25} e^{-in(\tau-\sigma)} + \bar{\alpha}_n^{25} e^{-in(\tau+\sigma)}\right)\,, \end{aligned} \tag{7.3}$$

which can be split into left and right movers $X^{25}(\sigma,\tau) = X_L^{25}(\sigma-\tau) + X_R^{25}(\sigma+\tau)$. x^{25} and p^{25} obey the usual commutation relations

$$\left[x^{25}, p^{25}\right] = i\,. \tag{7.4}$$

Because p^{25} is the generator of translations of x^{25}, one consequence of the quantization of the winding number is the quantization of the momentum mode p^i. This leads to the condition $p^i = \frac{M}{R}\,, \quad M \in \mathbb{Z}$. M is then called the momentum number. Additionally, we compute the mass formula

$$m^2 = \frac{M^2}{R^2} + \frac{L^2 R^2}{4} + (N_L + N_R - 2)\,, \tag{7.5}$$

where N_L and N_R count the number of oscillator modes. The first two components arise here from the compactified dimensions. Additionally, the compactified string has to fulfill the reparameterization condition (as the uncompactified string). This gives us the condition

$$N_R - N_L = M \cdot L \tag{7.6}$$

for the compactified string. Examining equation (7.5), we can see there exists a symmetry when exchanging winding modes L and momentum modes M. This transformation in total is given by

$$R \to \frac{2}{R}\,, \qquad L \leftrightarrow M\,. \tag{7.7}$$

This transformation leaves the level-matching condition and the mass spectrum unchanged. It can be shown that those two settings are indistinguishable not only on the level of the mass spectrum but on all levels and therefore, we can look at either a large or a small radius for the internal sector. Focusing on the massless sector of the compactified theory with $N_L + N_R = 2$, we find, due to equation (7.6), that $N_R = N_L = 1$, otherwise the string would not be massless. As a result the spectrum contains only a symmetric metric g_{ij}, an anti-symmetric two-form B_{ij} and a dilaton ϕ [62].

Focusing on the superstring and the type II string theories, we have to take additional effects into account. Here, T-dualities act as an anti-symmetric reflection of the right-moving sector and therefore the effect of on the Ramond sector, combined with the GSO-projections leads to a duality between Type IIA and Type IIB theory:

$$\text{Type IIB on } S^1_R \xleftrightarrow{\text{T}-\text{duality}} \text{Type IIA on } S^1_{2/R}\,. \tag{7.8}$$

This equivalency is true for higher-dimensional compactifications as well.

We restrict our attention on the bosonic sector of the theory, compactified on a d-dimensional torus with metric g_{ij}. We have to quantize the winding modes n^i and the momentum modes m_i in all compactified directions. For such a compactified string we find the mass formula [19]

$$m^2 = (N_L + N_R - 2) + \sum_{i,j=1}^{d} \left(m_i g^{ij} m_j + \frac{1}{4} n^i g_{ij} n^j \right) , \tag{7.9}$$

and as the level-matching condition

$$N_R - N_L = \sum_{i=1}^{d} m_i n^i\,. \tag{7.10}$$

As next step, we can add a Kalb-Ramond B-field $\mathbf{b}$. This shifts the mass momentum of the string and we get for the mass formula

$$m^2 =(N_L + N_R - 2) + \mathbf{m}^T \mathbf{g}^{-1} \mathbf{m} + \frac{1}{4} \mathbf{n} \left(\mathbf{g} - \mathbf{b} \mathbf{g}^{-1} \mathbf{b} \right) \mathbf{n} + \mathbf{n}^T \mathbf{b} \mathbf{g}^{-1} \mathbf{m} \pm \mathbf{n}^T \mathbf{m}\,. \tag{7.11}$$

Introducing a generalized momentum $\mathcal{P}^M$ and a generalized metric $\mathcal{H}_{MN}$

$$\mathcal{P}^M = \begin{pmatrix} m_i \\ n^j \end{pmatrix} , \qquad \mathcal{H}_{MN} = \begin{pmatrix} g^{ij} & -g^{ik} b_{kj} \\ b_{jk} g^{kj} & g_{ij} - b_{ik} g^{kl} b_{lj} \end{pmatrix} , \tag{7.12}$$

we can rewrite the mass formula and the level matching condition

$$\begin{aligned} m^2 &= (N_L + N_R - 2) + \mathcal{P}^M \mathcal{H}_{MN} \mathcal{P}^N , \\ N_R - N_L &= \frac{1}{2} \mathcal{P}_M \mathcal{P}^M\,. \end{aligned} \tag{7.13}$$

Here, we use as convention that $M, N = 1, \ldots, 2d$ and $i, j = 1, \ldots, d$. Indices are raised and lower with the $O(d, d)$-metric η_{MN}:

$$\eta_{MN} = \begin{pmatrix} 0 & \delta^i{}_j \\ \delta_i{}^j & 0 \end{pmatrix} , \ \eta^{MN} = \begin{pmatrix} 0 & \delta_i{}^j \\ \delta^i{}_j & 0 \end{pmatrix} , \ \eta^{MP} \eta_{PN} = \delta^M{}_N\,. \tag{7.14}$$

The generalized metric $\mathcal{H}_{MN}$ fulfills the properties of the $O(d,d)$ group, and therefore, the following identities hold:

$$\mathcal{H}^{MN}\mathcal{H}_{NP} = \eta_{MP}\,, \qquad \mathcal{H}^T\eta\mathcal{H} = \eta\,, \qquad \mathcal{H}^{MN} = \eta^{MP}\eta^{NQ}\mathcal{H}_{PQ}\,. \tag{7.15}$$

Because of the underlying $O(d,d)$-symmetry, we can generalize the T-duality from (7.7) to the formulae

$$\mathcal{H}_{MN} \leftrightarrow h_M{}^P\mathcal{H}_{PQ}h_N{}^Q\,, \qquad \mathcal{P}^M \leftrightarrow h^M{}_N\mathcal{P}^N\,, \qquad h \in O(d,d\mathbb{Z})\,. \tag{7.16}$$

Note, that the element h must be an element of $O(d,d,\mathbb{Z})$, and not of $O(d,d,\mathbb{R})$ because the generalized momenta must be still quantized after a T-duality transformation.

The basis of $O(d,d)$ consists of three different basis elements:

- diffeomorphisms (basis transformations of the torus), which only act on n^i and p_i separately given by:

$$h_M{}^N = \begin{pmatrix} E^i{}_j & 0 \\ 0 & E_i{}^j \end{pmatrix}\,, \qquad\qquad E \in \mathrm{GL}(d) \tag{7.17}$$

- shifts given by

$$h_M{}^N = \begin{pmatrix} \delta^i{}_j & 0 \\ n_{ij} & \delta_i{}^j \end{pmatrix}\,, \qquad\qquad n_{ij} = -n_{ji} \tag{7.18}$$

- and the T-duality transformations in the k-th direction

$$h_M{}^N = \begin{pmatrix} \delta^i{}_j - t^i{}_j & t^{ij} \\ t_{ij} & \delta_i{}^j - t_i{}^j \end{pmatrix}\,, \qquad t_{ij} = \mathrm{diag}\,(0,\dots 0,1,0\dots 0) \tag{7.19}$$

The transformation rules for the T-duality transformation for the metric g_{ij} and for the two-form b_{ij} are called Buscher rules [63, 64]:

$$\begin{aligned} g_{kk} \to \frac{1}{g_{kk}}\,,\ g_{ki} \to \frac{b_{ki}}{g_{kk}}\,,\ g_{ij} \to g_{ij} - \frac{g_{ki}g_{kj} - b_{ki}b_{kj}}{g_{kk}}, \\ b_{ki} \to \frac{g_{ki}}{g_{kk}}\,,\ b_{ij} \to b_{ij} - \frac{g_{ki}b_{kj} - b_{ki}g_{kj}}{g_{kk}}\,. \end{aligned} \tag{7.20}$$

Having such an underlying duality in string theory motivated the construction of double field theory (DFT), a supergravity theory which is manifestly invariant under T-duality. For a review of double field theory, we refer to [62]. In chapter 9, we are going to use DFT implicitly, but the section can be understood without any knowledge of DFT.

7.1.2 S-duality

Another important duality in string theory is the S-duality of Type IIB string theory, and is therefore also a duality of M-theory as we can see in Figure 1.1. We are going to look at the bosonic part of Type IIB string theory. The NS-NS sector is defined by the dilaton ϕ, the metric g_{ij} and the Kalb-Ramond field B, the R-R sector contains the C_0-form, the C_2-form and the self-dual C_4-form. The basic idea of the duality is the map

$$g_s \to 1/g_s\,. \tag{7.21}$$

This duality acts as a $\mathrm{SL}(2,\mathbb{Z})$ transformation on the R-R sector of the type IIB string, the p-forms C_0, C_2 and C_4, as well as on the NS-NS sector. Combining the dilaton (which is connected to the string coupling via $g_s = e^{\phi}$) and the scalar C_0, we find that it transforms as a complex scalar under a modular transformation:

$$\tau = C_0 + ie^{-\phi}\,, \qquad \tau \to \frac{a\tau + b}{c\tau + d}\,. \tag{7.22}$$

We can organize the SL(2) parameters into a matrix with unit determinant:

$$U = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \qquad \det U = ad - bc = 1\,. \tag{7.23}$$

This $SL(2)$-transformation acts on the matrix as

$$\mathcal{M} = \frac{1}{\mathrm{Im}\tau}\begin{pmatrix} \|\tau\|^2 & \mathrm{Re}\tau \\ \mathrm{Re}\tau & 1 \end{pmatrix} = e^{\phi}\begin{pmatrix} C_0^2 + e^{-2\phi} & C_0 \\ C_0\tau & 1 \end{pmatrix}, \tag{7.24}$$

the transformation itself can be written as $\mathcal{M} \to U\mathcal{M}U^T$. Setting $a = d = 0$ and $b = -c = 1$, we end up with the inversion of the string coupling.

The two two-forms C_2 and B transforming as a doublet under

$$\begin{pmatrix} C_2 \\ B \end{pmatrix} \to U \begin{pmatrix} C_2 \\ B \end{pmatrix}, \tag{7.25}$$

while the four-form C_4 is invariant, as well the Einstein frame, which is connected to the string frame by the formula $g_{ij}^{\text{string}} = e^{\phi/2} g_{ij}^{\text{Einstein}}$.

7.1.3 U-duality

As next step, we want to combine T- and S-duality, as it is suggested by Figure 1.1 that both should be underlying symmetries to a possible formulation of M-theory. This unification is called U-duality. We can combine these two dualities in the framework of Type II string theories. The basic idea is to combine two T-dualities with one S-duality as

$$\text{Type IIA} \xleftrightarrow{\text{T-duality}} \text{Type IIB} \xleftrightarrow{\text{S-duality}} \text{Type IIB} \xleftrightarrow{\text{T-duality}} \text{Type IIA}\,. \tag{7.26}$$

To unify the notation of those dimensions it is suitable to decompactify the Type IIA string theory on a circle and relate the radius R_{11} and the Planck length ℓ_p to the string coupling g_s and the string length ℓ_s as

$$R_{11} = \ell_s g_s \,, \qquad \ell_p = g_s^{1/3} \ell_s \,. \tag{7.27}$$

Then, the 11-dimensional metric can be written as

$$ds_{11}^2 = R_{11}^2 \left(dx^{11} + A \right)^2 + ds_{10}^2 \,, \tag{7.28}$$

where A is a 10-dimensional one-form. Using the connection between the string coupling g_s and the dilaton e^{ϕ} at asymptotic infinity, we find the general ansatz

$$\frac{1}{\ell_p^2} ds_{11}^2 = e^{4\phi/3} \left(dx^{11} + A \right)^2 + e^{-2\phi/3} \frac{1}{\ell_s^2} ds_{10}^2 \,. \tag{7.29}$$

To generate 11-dimensional U-dualities, we have to rewrite the T-dualities and S-dualities in terms of 11-dimensional quantities. Using these uplifts, we find the following symmetrical U-duality acting in three different directions

$$U_{IJK} \,:\, R_I \to \frac{\ell_p^3}{R_J R_K} \,, \qquad \ell_p \to \frac{\ell_p^2}{\sqrt[3]{R_I R_J R_K}} \,, \tag{7.30}$$

where I, J and K are distinct and can be any of $(i, j, 11)$. This transformation is an inversion of the volume $V = R_I R_J R_K$, for the dimensionless quantity V/ℓ_p^3 and we find $V/\ell_p^3 = \tilde{V}/\tilde{\ell}_p^3$.

The transformation can be rephrased in terms of dimensionless quantities by

$$U_{IJK} \,:\, \frac{R_I}{\ell_p} \to \frac{\ell_p R_I^{1/3}}{\left(R_J R_K \right)^{2/3}} \,. \tag{7.31}$$

This allows us to set $\ell_p = 1$ and therefore we can use (7.31) to define the U-duality transformation in the absense of any other fields. Then, the volume transformation become $\tilde{V} = 1/V$, where $\tilde{V}$ is the transformed volume. More general transformations involving all possible fields are presented in section 7.2.3.

We are now going to look at some explicit low-dimensional examples, and assume the extension to higher-dimensional U-duality groups. Let us start looking at M-theory[1] on a two-dimensional torus. Symmetries of the equations of motion are groups which preserves the torus itself, but are allowed to change the complex structure parameter τ. Therefore, the symmetry is $\mathrm{SL}(2, \mathbb{R})$. Additionally, global scaling symmetries are allowed as well. In two dimension this corresponds to $\mathbb{R}^+$. Therefore, the complete symmetry group is $\mathrm{SL}(2, \mathbb{R}) \times \mathbb{R}^+$.

For M-theory on a three-dimensional torus, we described a subset of possible transformations in (7.31). In general, one would expect the full $\mathrm{SL}(3, \mathbb{R})$, including shift symmetries

---

[1] or rather eleven-dimensional supergravity

of the the three-form $C_{IJK} \to C_{IJK} + \ell_p^3 \Lambda_{IJK}$ with Λ_{IJK} integer-valued. Additionally, we have a scaling symmetry. This transformation acts on the volume of the torus as well as the single component of the background three-form C_{123}. We can introduce an additional complex structure constant with

$$\tau = \frac{C_{123}}{\ell_p^3} + i \frac{\sqrt{\det g}}{\ell_p^3} \,, \tag{7.32}$$

where g_{ij} is the metric. In this transformation also the shift of the three-form is included, when taking $a = d = b = 1$ and $c = 0$.

Therefore, the complete symmetry group of the 3-dimensional symmetry group is $\mathrm{SL}(3,\mathbb{R}) \times \mathrm{SL}(2,\mathbb{R})$. In classical supergravity theory these groups are real-valued, but it is expected that they are integer-valued in a full quantum theory. We can write down the matrices explicitly where the transformation acts on. The transformation SL(3) acts on

$$\mathcal{M}_{ij} = \frac{1}{(\det g)^{1/3}} g_{ij} \,, \tag{7.33}$$

which can be interpreted as coset elements $\mathcal{M}_{ij} \in SL(3)/SO(3)$ where $i, j = 1, 2, 3$. The second part can be written as

$$\mathcal{M}_{\alpha\beta} = \frac{1}{(\det g)^{1/2}} \begin{pmatrix} \ell_p^{-3} \left(C_{123}^2 + \det g\right) & C_{123} \\ C_{123} & \ell_p^3 \end{pmatrix} , \tag{7.34}$$

where $\alpha, \beta = 1, 2$ and $\mathcal{M}_{\alpha\beta}$ is a coset element of SL(2)/SO(2). Combining these two matrices we get a single six-by-six matrix

$$\mathcal{M}_{MN} \equiv \mathcal{M}_{i\alpha, j\beta} = \mathcal{M}_{ij} \mathcal{M}_{\alpha\beta} \,, \tag{7.35}$$

where we split the index $M = i\alpha$. In higher dimensions, more shift symmetries are possible and has to be included. In the papers [65, 66, 67] it was shown that the dimensional reduction of eleven-dimensional supergravity on a d-dimensional torus leads to the symmetry group $\mathrm{E}_{d(d)}$. The different groups and some of their properties can be found in table 7.1.

7.2 Exceptional Field Theory

After an in-depth analyses of the different dualities and their symmetry groups, we will give a short review on exceptional field theory (EFT) which is manifestly invariant under U-duality. To construct such a theory, we have to find the generalized diffeomorphisms which generate infinitesimal gauge transformations. With these generalized diffeomorphisms we can construct the generalized metric (and the generalized vielbein) and then find the action from this metric. As last step, we are going to compute the transformation behaviour under general U-dualities for the different fields and construct duality chains to find non-geometric backgrounds.

d	$\mathrm{E}_{d(d)}$	H_d	$\mathbf{R_1}$	**adj.**
3	$\mathrm{SL}(3) \times \mathrm{SL}(2)^+$	$\mathrm{SO}(3) \times \mathrm{SO}(2)$	$(\mathbf{3}, \mathbf{2})$	$(\mathbf{8}, \mathbf{1}) \oplus (\mathbf{1}, \mathbf{3})$
4	$\mathrm{SL}(5)$	$\mathrm{SO}(5)$	$\mathbf{10}$	$\mathbf{24}$
5	$\mathrm{SO}(5,5)$	$\mathrm{SO}(5) \times \mathrm{SO}(5)$	$\mathbf{16}$	$\mathbf{45}$
6	$\mathrm{E}_{6(6)}$	$USp\,(8)$	$\mathbf{27}$	$\mathbf{78}$
7	$\mathrm{E}_{7(7)}$	$\mathrm{SU}(8)$	$\mathbf{56}$	$\mathbf{133}$
8	$\mathrm{E}_{8(8)}$	$\mathrm{SO}(16)$	$\mathbf{248}$	$\mathbf{1} \oplus \mathbf{3875}$

Figure 7.1: The exceptional groups in the dimensions $3 \leq d \leq 8$, their maximal compact subgroups (ignoring discrete factors) and their R_1 representation together with the adjoint. Note that the generalized vectors transform in R_1, while the generalized derivatives transforms in $\overline{R}_1 = R_{8-d}$. This is not well defined for $d = 8$, and therefore the dimension $d = 8$ is not accessible by the presented framework.

7.2.1 Generalized Diffeomorphisms

We are interested in the local symmetries of our system. Therefore, we are going to define generalized diffeomorphisms which include diffeomorphisms and gauge transformation which are realized as a local $\mathrm{E}_{d(d)}$ action on the fields by the generalised Lie derivative. As a first step, we start to introduce generalized vectors. In general, they have the same dimensionality as the fundamental group of the exceptional fields and, comparing to (7.5) label the momentum and the different wrapping modes of the membranes. These generalized vectors are in generalized geometry seen as a section of exceptional tangent bundles

$$\mathcal{R}_1 \simeq TM_d \oplus \Lambda^2 T^* M_d \oplus \Lambda^5 T^* M_d \oplus \tag{7.36}$$

where M_d is the internal space. Therefore, generalized vectors are split under these representation into

$$V^M = \left(v^i, v_{ij}, v_{ijklm}\right) , \tag{7.37}$$

with $i, j, ... = 1, ..., d$ for $d < 7$. For $d = 7$, we have to take additionally mixed-symmetry terms into account

$$\mathcal{R}_1 \simeq TM_d \oplus \Lambda^2 T^* M_d \oplus \Lambda^5 T^* M_d \oplus \left(T^* M_d \otimes \Lambda^7 T^* M_d\right) . \tag{7.38}$$

This additional section is a consequence of the four remaining non-compactified dimensions. Due to the self-duality of one-forms we get a “dual graviton”, which then leads to an additional gauge parameter

Those generalized vectors lead to corresponding generalized derivatives ∂_M with respect to a generalized set of coordinates $Y^M \in \mathcal{R}_1$, we can write them as:

$$\begin{aligned}\partial_M &= \left(\partial_i, \partial^{ij}, \partial^{ijklm}, \partial^{ijklmno,p}\right) \,, \\ Y^M &= \left(x^i, w_{ij}, w_{ijklm}, w_{ijklmno,p}\right) \,,\end{aligned} \tag{7.39}$$

with $i, j, ... = 1, ..., d$. Here, all indices are anti-symmetric. Viewing x^i as position in the compactified space, the coordinate $w_{ij}, w_{ijklm}, w_{ijklmno,p}$ corresponds to the wrapping modes of different types of branes: the coordinate w_i corresponds to M2-branes, w_{ijklm} to M5-branes and $w_{ijklmno,p}$ to KKM-wrapping modes. Therefore, the new appearance in the exceptional tangent bundles corresponds to additional allowed wrapping modes in the internal sector of exceptional field theory.

The action on a vector field in the $\mathcal{R}_1$ representation, $V \in \Gamma\left(\mathcal{R}_1\right)$ with weight λ_V is

$$\mathcal{L}_\Lambda V^M = \Lambda^N \partial_N V^M - V^N \partial_N \Lambda^M + Y^{MP}_{NQ} V^N \partial_P \Lambda^Q + \left(\lambda_V + \omega_d\right) V^M \partial_N \Lambda^N \,. \tag{7.40}$$

Here, $\partial_M = \frac{\partial}{\partial X^M}$ are the derivatives corresponding to the generalized coordinates, and the Y-tensor is defined using a projector on the adjoint representation:

$$Y^{MN}{}_{PQ} = -\alpha_d \mathbb{P}^M{}_Q{}^N{}_P + \delta^M{}_P \delta^N{}_Q - \omega_d \delta^M{}_Q \delta^N{}_P \,. \tag{7.41}$$

The constant ω_d is given by

$$\omega_d = -\frac{1}{9-d} \,, \tag{7.42}$$

and α_d is a numerical constant. The exact structure of Y-tensors for any dimension d was computed in [68], and corresponds to the invariants of each symmetry group. The structure for the individual dimensions can be found in appendix A.1.

For the weights of the objects, we have to pick a convention. We are going to use the convention that a vielbein should have determinant one, and should be an unweighted tensor, meaning that

$$E^{\bar{M}}{}_M \mathcal{L}_V E_{\bar{M}}{}^M = \mathcal{L}_V |E| = 0 \,. \tag{7.43}$$

Therefore, the metric is an unweighted tensor as well, using the definition

$$\mathcal{M}_{MN} = E^{\bar{M}}{}_M E^{\bar{N}}{}_N \delta_{\bar{M}\bar{N}} \,. \tag{7.44}$$

Additionally, we choose the weights of a generalized vector V^M to be $\lambda = -\omega$, and therefore, the last term in (7.40) vanishes.

Demanding the closure of the algebra, i.e.

$$\left[\mathcal{L}_U, \mathcal{L}_V\right] W = \mathcal{L}_{[U,V]_E} W \,, \qquad [U, V] \equiv \frac{1}{2}\left(\mathcal{L}_U V - \mathcal{L}_V U\right) \,, \tag{7.45}$$

where U, V and W are generalized vectors of weight $-w$, we find that we have to impose the condition

$$Y^{MN}{}_{PQ} \partial_M \otimes \partial_N = Y^{MN}{}_{PQ} \partial_M \partial_N = 0 \,, \tag{7.46}$$

where $\otimes$ represents the action of the derivatives on two independent fields. This condition is called the section condition of Exceptional Field Theory. From a physical point of view this condition can be seen as an extension of level matching condition to neglect additional degrees of freedom, arising from the extended theory.

7.2.2 Action of EFT

After defining the generalized diffeomorphisms, we are going to define the action in $d = 4$. Actions behave like a space time scalars under generalized diffeomorphisms and are quadratic in the derivatives. To construct such an action, we write the generalized metric in four dimensions:

$$\mathcal{M}_{MN} = \begin{pmatrix} m_{i,k} & m_{i,kl} \\ m_{ij,k} & m_{ij,kl} \end{pmatrix} = |g|^{\frac{1}{5}} \begin{pmatrix} g_{ik} + \frac{1}{2} C_{imn} C_k{}^{mn} & \frac{1}{2} C_i{}^{mn} \epsilon_{klmn} \\ \frac{1}{2} C_k{}^{mn} \epsilon_{ijmn} & |g| g_{i[k} g_{l]j} \end{pmatrix} . \tag{7.47}$$

We can now construct the action using where Δ is the determinant of the seven-dimensional external metric. Then, the action is

$$\begin{aligned} S = \int d^{10}X \; e^{-2\Delta} \Big(& \frac{1}{12} \mathcal{M}^{MN} \partial_M \mathcal{M}^{KL} \partial_N \mathcal{M}_{KL} - \frac{1}{2} \mathcal{M}^{MN} \partial_M \mathcal{M}^{KL} \partial_K \mathcal{M}_{LN} \\ & + \frac{24}{7} \left(\partial_M \mathcal{M}^{MN} \partial_N \Delta - \mathcal{M}^{MN} \partial_M \Delta \partial_N \Delta \right) \\ & - \partial_M \partial_N \mathcal{M}^{MN} \Big) . \end{aligned} \tag{7.48}$$

Using the section condition $\partial^{ij} = 0$, this action truncates to the bosonic part of eleven-dimensional supergravity assuming that all fields depend only on the four internal coordinates. Therefore, we have been able to successfully construct an action with an underlying U-duality. This action has been found in [69]. The complete action for Exceptional Field Theory was computed for every dimension individually in the papers [70, 71, 72, 73, 74, 75, 76, 77].

7.2.3 U-dualities

Later, we have to compute the effect of U-dualities explicitly on twisted tori and backgrounds with three-forms and six-forms. Those were computed in [78].

We are not only going to have a geometric three-form C_{ijk} and a geometric six-form C_{ijklmn}, but also non-geometric three-forms Ω^{ijk} and non-geometric six-forms Ω^{ijklmn}. These two types of forms arise from generalized geometry, and are two different types of descriptions for the background. For many backgrounds, one of these descriptions might be ill-defined, as long as the other is well-defined the background can be viewed as a well-defined background. In four dimensions, the transformation between the two parameterizations are given by

$$\begin{aligned} \hat{g}_{ij} &= \left(1+V^2\right)^{-1/3}\left[\left(1+V^2\right)g_{ij} - V_i V_j\right] , \\ \Omega^{ijk} &= \left(1+V^2\right)^{-1} g^{il} g^{jm} g^{kn} C_{lmn} , \\ \hat{ds}^2_7 &= \left(1+V^2\right)^{-1/3} ds^2_7 . \end{aligned} \tag{7.49}$$

Here, the $V^i = \frac{1}{3!|e|}\epsilon^{ijkl}C_{jkl}$ with $\epsilon^{ijkl} = \pm 1$ is the tensor density and $V^2 = V^i V^j g_{ij}$.

To describe backgrounds, we use the already introduced generalized vielbeins Note that there exist a geometric vielbein E_C and a non-geometric vielbein E_Ω. For $d = 4$ they are

$$E_C{}^{\bar{a}}{}_a = \begin{pmatrix} |e|^{-2/5} e^{\bar{i}}{}_i & \frac{1}{3!}|e|^{-2/5}\epsilon^{\bar{i}jkl}C_{jkl} \\ 0 & |e|^{3/5} \end{pmatrix} , \qquad E_\Omega{}^{\bar{a}}{}_a = \begin{pmatrix} |e|^{-2/5} e^{\bar{i}}{}_i & 0 \\ \frac{1}{3!}|e|^{3/5}\epsilon_{ijkl}\Omega^{jkl} & |e|^{3/5} \end{pmatrix} , \tag{7.50}$$

where ϵ_{ijkl} is the four-dimensional Levi-Civita symbol such that $\epsilon_{1234} = 1$. Using those vielbein, we can now construct rules for U-dualities using

$$U = E_\Omega \; E_C \; E_\Omega \, . \tag{7.51}$$

For U-dualities in the three directions i, j and k, we define the anti-symmetric combinations

$$\omega_{ijk} = \omega^{ijk} = \pm 1 , \tag{7.52}$$

which only have non-vanishing components along the three directions being dualized. Then, we construct the generalized vielbein with the (non)-geometric three-form using

$$\Omega^{ijk} = -\omega^{ijk} , \qquad C_{ijk} = \omega_{ijk} . \tag{7.53}$$

We find the analogous case for the U-dualities in six dimensions. For a U-duality in the direction i, j, k, l, m and n, we define the anti-symmetric quantity

$$\omega_{ijklmn} = \omega^{ijklmn} = \pm 1 , \tag{7.54}$$

and define the (non-)geometric vielbein with the three-forms

$$\Omega^{ijklmn} = -\omega^{ijklmn} , \qquad C_{ijklmn} = \omega_{ijklmn} . \tag{7.55}$$

Using those transformation matrices, we can compute the new three-forms and six-forms as well as the exchanged coordinates, momenta and wrapping modes. The explicit formulae can be found in appendix A.2.

7.3 Duality Chains

As final step, we are going to review duality chains, one with applying T-dualities, the other with U-dualities. In the following chapter 8, we are going to use those to obtain information on the phase space of non-geometric backgrounds. Note that here in this section we focus only on toy models with only one type of flux present, but more complicated backgrounds can be constructed (see for instance [79]).

7.3.1 Duality chains for T-dualities

In [80], the concept of duality chains leading to non-geometric background fields was first examined. Starting from geometric background with a T^3 with H-flux, we use the Buscher rules (7.20) to compute the dual backgrounds.

T^3 with H-flux: The metric of the T and the H-flux $H_{123} = 3\partial_{[1}B_{23]}$ can be written as

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 , \\ B_{xy} &= Nx^3 , \end{aligned} \tag{7.56}$$

where $N \in \mathbb{Z}$ due to quantization conditions on the torus.

Twisted torus: Applying the Buscher rules described in the x^1-direction, we obtain a nilmanifold with

$$\begin{aligned} ds^2 &= \left(dx^1 - Nx^3 dx^2\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 , \\ B &= 0 . \end{aligned} \tag{7.57}$$

Therefore, this background has the "geometric flux" ${T_{23}}^1 = N$. More details on this type of space can be found in section 8.2.1. This configuration is also called twisted torus because of the identification

$$(x^1, x^2, x^3) \sim (x^1 + 1, x^2, x^3) \sim (x^1, x^2 + 1, x^3) \sim (x^1 + Nx^2, x^2, x^3 + 1) . \tag{7.58}$$

Q-flux background: The first duality was along the x^1 direction, which is an isometry because the map $x^1 \to x^1 + 1$ is a map onto itself. The next direction to dualize is the x^2 direction. After this duality we get an ill-defined background given by

$$\begin{aligned} ds^2 &= \frac{(dx^1)^2 + (dx^2)^2}{1 + (Nx^3)^2} + \left(dx^3\right)^2 , \\ B_{\tilde{x^1}\tilde{x^2}} &= -\frac{Nx^3}{1 + (Nx^3)^2} . \end{aligned} \tag{7.59}$$

Both of these fields are not-well defined under the transformation $x^3 \to x^3 + 1$ in conventional geometry but can be patched by an SO(3, 3) element. Such a background is called T-fold. Because it is not accessible with ordinary geometry, the background is non-geometric whereas in generalized geometry or double field theory this is well-defined. Here, we express the background in terms of the bi-vector β^{ij} which is well-defined. To derive this, we use the redefinition known from[81, 82, 83, 84, 85, 86, 87]

$$\begin{aligned} \beta^{ij} &= \frac{1}{2}\left((g - B)^{-1} - (g + B)^{-1}\right) , \\ \hat{g} &= \frac{1}{2}\left((g - B)^{-1} + (g + B)^{-1}\right)^{-1} . \end{aligned} \tag{7.60}$$

As in exceptional field theory, this non-geometric description can be viewed as a different parameterization of the generalized vielbein. The resulting background is

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 \,, \\ \beta^{12} &= N x^3 \,. \end{aligned} \tag{7.61}$$

This non-geometric background is classified by the Q-flux, which is given by

$$Q_i{}^{jk} = \partial_i \beta^{jk} \,. \tag{7.62}$$

For the given background the only non-zero component is

$$Q_3{}^{12} = N \tag{7.63}$$

R-flux background: As last step, we perform a T-duality in the x^3-direction. Even though this is no isometry direction, we can use it from a double field theory point of view and directly apply the Buscher rules applying the generalized transformation. This results in an exchange of the coordinate x^3 by its dual mode $\tilde{x}_3$ which corresponds to an exchange between winding and momentum modes. This results in the background:

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 \,, \\ \beta^{12} &= N \tilde{x}_3 \,. \end{aligned} \tag{7.64}$$

The R-flux which is given by

$$R^{ijk} = 3\partial^{[i}\beta^{jk]} \,, \tag{7.65}$$

which results in the given background in the R-flux

$$R^{123} = N \,. \tag{7.66}$$

This is a realization of a non-geometric background. In the following, we find similar results in exceptional field theory.

7.3.2 Duality chains for U-dualities

As next step, we uplift the duality chain to M-theory, as we did before to perform U-dualities. Because we always have to perform two T-dualities (and a corresponding S-duality in another direction), we are going to split the duality chain into two parts, namely

$$H_{1234} \xrightarrow{U_{124}} Q_3{}^{124} \qquad \text{and} \qquad T^1{}_{23} \xrightarrow{U_{234}} R^{4,1234} \tag{7.67}$$

Duality chain $H_{1234} \xrightarrow{U_{124}} {Q_3}^{124}$

We start with a three-dimensional torus times a circle $T^3 \times S^1$. The additional circle is the circle for the M-theory. Compactifying this direction leads to the known fluxes in string theory. We start with the configuration:

$$\begin{aligned} ds^2 =& \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 \,, \\ C_{124} =& N x^3 \,. \end{aligned} \tag{7.68}$$

Therefore, we have the flux $H_{1234} = 4\partial_{[1}C_{234]} = N$. Performing a U-duality in the directions 1, 2 and 4, and using the equations in appendix A.2, we find

$$\begin{aligned} ds^2 =& \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 \,, \\ \Omega^{124} =& N x^3 \,. \end{aligned} \tag{7.69}$$

This background has the flux ${Q_3}^{124} = \partial_3 \Omega^{124} = N$, which is a spacetime tensor as we later show in the next chapter 8.

Note that those formulae enable us to directly find the correct background description (geometric or non-geometric) compared to the previous section, where we performed the dualities explicitly and find a ill-defined background, which had to be patched by a change of the used description.

Compactifying the forth dimension leads us directly to the string theory fluxes:

$$\begin{aligned} C_{124} &\to B_{12} \\ H_{1234} &\to H_{123} \end{aligned} \tag{7.70}$$

The same is true for the Q-flux background:

$$\begin{aligned} \Omega^{124} &\to \beta^{12} \\ {Q_3}^{124} &\to {Q_3}^{12} \,. \end{aligned} \tag{7.71}$$

Therefore, this duality chain is a true uplift of the duality chain from string theory.

Duality chain $T_{23} \xrightarrow{U_{234}} R^{4,1234}$

For this duality chain, we have to start with a twisted torus. We use a twisted torus (as in string theory) times a circle as $\tilde{T}^3 \times S^1$. Here, we have the background

$$\begin{aligned} ds^2 =& \left(dx^1 - N x^3\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 \,, \\ C_{ijk} =& 0 \,. \end{aligned} \tag{7.72}$$

Performing a U-duality in the directions x^2, x^3 and x^4, we find for the background

$$\begin{aligned} ds^2 =& \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 \,, \\ \Omega^{134} =& N w_{24}, . \end{aligned} \tag{7.73}$$

This background has the non-geometric R-flux $R^{4,1234} = 4\hat{\partial}^{4,[1}\Omega^{234]} = N$, known from [88], where we used that $\hat{\partial}^{ij} = \partial^{ij} + \Omega^{ijk}\partial_k$. Again, we compactify along the x^4-direction and find the known string fluxes. In chapter 8 we are going to extend these duality chains in exceptional field theory to construct backgrounds with other non-geometric fluxes. We are going to use the framework of generalized diffeomorphisms to be able to find their structure.

Chapter 8

Locally Non-Geometric Fluxes and Missing Momenta in M-theory

This chapter focuses on the results of paper [25]. We use the framework of exceptional field theory to describe geometric and globally/locally non-geometric fluxes of M-theory. These fluxes can be found using the decomposition of the fundamental groups. Analyzing them leads to mixed symmetry tensors which describe new R-R fluxes in the weak coupling regime. We examine the structure of the decompositions of the symmetry groups of the embedding tensor of SUGRA and find the structure of the fluxes. This will provide us insight which we can utilize to study scenarios beyond the NS-NS case, while showcasing the power of analyzing symmetry groups in string theory. Finally, we are going to focus on the construction of duality chains which lead to these new non-geometric backgrounds and show that certain momenta associated with the non-geometric fluxes vanish.

8.1 Fluxes in M-Theory

8.1.1 Review of four-dimensional backgrounds

We start our analysis of the four-dimensional case reviewing and extending the results from [88]. Here, the exceptional group is the SL(5)-group. As we know from the review in section 7.1.3, the R_1-representation is ten-dimensional (see table 7.1). A canonical labeling of the coordinates within this ten-dimensional space is $X^M = X^{[ab]} \equiv X^{ab}$, with $a, b = 1, \dots, 5$, which can be further split into the four-dimensional representation $X^{ab} = (X^{i5}, X^{ij}) = \left(x^i, \frac{1}{2}\epsilon^{ijkl} w_{ij}\right)$ with $i, j, \dots = 1, \dots, 4$. Here, x^i describes the spacetime coordinates, whereas $\tilde{x}_{ij}$ are the wrapping coordinates of the M2-brane. This can be also seen using the decomposition of $\mathbf{10}$ under $\mathrm{SL}(4) \times \mathbb{R}^+ \subset \mathrm{SL}(5)$

$$\mathbf{10} \longrightarrow \mathbf{4}_3 \oplus \overline{\mathbf{6}}_{-2} \,. \tag{8.1}$$

Here, we can identify the coordinates from the representation

$$x^i \in \mathbf{4}_3 \,, \qquad w_{ij} \in \overline{\mathbf{6}}_{-2} \,, \tag{8.2}$$

and therefore, also their derivatives contained in $\overline{\mathbf{10}}$

$$\partial_i \in \overline{\mathbf{4}}_{-3}\,, \qquad \partial^{ij} \in \mathbf{6}_2\,. \tag{8.3}$$

Using the decomposition of the adjoint representation

$$\mathbf{24} \longrightarrow \mathbf{15}_0 \oplus \overline{\mathbf{4}}_5 \oplus \mathbf{4}_{-5} \oplus \mathbf{1}_0\,, \tag{8.4}$$

we can directly identify the fields appearing in the generalized non-geometric vielbein of the SL(5) theory:

$$e_{\bar{i}}{}^i \in \mathbf{15}_0 \oplus \mathbf{1}_0\,, \qquad C_{ijk} \in \mathbf{4}_{-5}\,, \qquad \Omega^{ijk} \in \overline{\mathbf{4}}_5\,. \tag{8.5}$$

Let us note that the $\mathbb{R}^+$-charge gives us a measure of the "non-geometricness" of an object. Furthermore, the total $\mathbb{R}^+$-charge of a product of two fields is the sum of each individual $\mathbb{R}^+$-charge of each field.

Having defined the field content we continue with the generalized vielbein

$$E^{\bar{a}}{}_a = \begin{pmatrix} |e|^{-2/5} e^{\bar{i}}{}_i & 0 \\ \frac{1}{3!}|e|^{3/5}\epsilon_{ijkl}\Omega^{jkl} & |e|^{3/5} \end{pmatrix}\,, \tag{8.6}$$

where ϵ_{ijkl} is the four-dimensional Levi-Civita symbol such that $\epsilon_{1234} = 1$ and $|e| = \det e^{\bar{i}}{}_i$. The weights are determined by the generalized diffeomorphisms of the vielbein. All examples in this chapter are backgrounds which are generalized parallelisable, hence the inverse of the generalized vielbein defines a set of globally well-defined generalized vector fields. It will be straight forward to decide which parameterization leads to a globally well-defined vielbein. For a change of the description, we can use the formulae given in (7.49).

Construction of non-geometric fluxes

In this thesis, we want to give insights in how to find new fluxes and how to check the corresponding fluxes whether they are spacetime tensors. Therefore, we present the case of SL(5) exceptional field theory with many details to give insights in how to find the correct fluxes, the generalization to higher dimensions is shorter and focus only on new arising problems.

To start the search, we are first going to identify the transformation behavior of the individual fields and derivatives under generalized diffeomorphisms and afterwards construct the different fluxes. Note, that we are interested in spacetime tensors, and therefore restrict the generalized diffeomorpisms of section 7.2.1, that only the spacetime diffeomorphism generator is non-zero

$$V^{ab} = \left(V^{i5},\, V^{ij}\right) \longrightarrow \left(\xi^i,\, 0\right)\,, \tag{8.7}$$

which leads to a simplification of the generalized diffeomorphism

$$\mathcal{L}_V \rightarrow \delta_\xi\,. \tag{8.8}$$

Applying this on the generalized vielbein we find for the non-geometric three-form

$$\begin{aligned}\delta_\xi \Omega^{ijk} &= \xi^l \partial_l \Omega^{ijk} - 3\Omega^{l[ij}\partial_l \xi^{k]} - 3\partial^{[ij}\xi^{k]}\,,\\ &= L_\xi \Omega^{ijjk} - 3\partial^{[ij}\xi^{k]}\,,\end{aligned} \tag{8.9}$$

where L_ξ is the usual spacetime Lie derivative acting on a tensor. Additionally, we find that the metric $g_{ij} = e^{\bar{i}}{}_i e_{\bar{i}j}$ is a spacetime tensor

$$\delta_\xi g_{ij} = L_\xi g_{ij} = \xi^k \partial_k g_{ij} + 2g_{k(i}\partial_{j)}\xi^k\,, \tag{8.10}$$

which is expected from the definition of the generalized diffeomorphism. The three-form C_{ijk} also transforms as a spacetime tensor, i.e.

$$\delta_\xi C_{ijk} = L_\xi C_{ijk}\,. \tag{8.11}$$

As next step, we examine the transformation behavior of the derivatives (acting on a scalar). The ordinary derivative ∂^i transforms as a spacetime tensor, whereas the derivative of the winding modes does not, i.e.

$$\begin{aligned}\delta_\xi\left(\partial_i\phi\right) =& \hat{L}_\xi\left(\partial_i\phi\right)\,,\\ \delta_\xi\left(\partial^{ij}\phi\right) =& \hat{L}_\xi\left(\partial^{ij}\phi\right) - 3\partial^{[ij}\xi^{k]}\,.\end{aligned} \tag{8.12}$$

To counter this, we introduce an improved derivative $\hat{\partial}$ for the winding modes such that it becomes a spacetime tensor. This improved derivative is given by

$$\hat{\partial}^{ij} = \partial^{ij} + \Omega^{ijk}\partial_k\,. \tag{8.13}$$

We notice that this improved derivative transforms as a spacetime tensor, i.e.

$$\delta_\xi\left(\hat{\partial}^{ij}\phi\right) = \hat{L}_\xi\left(\hat{\partial}^{ij}\phi\right) + \partial^{ik}\phi\partial_k\xi^j + \partial^{ik}\xi^j\partial_k\phi - \partial^{jk}\phi\partial_k\xi^i - \partial^{jk}\xi^i\partial_k\phi = \hat{L}_\xi\left(\hat{\partial}^{ij}\phi\right)\,. \tag{8.14}$$

In the previous computation we have to use the section condition, which in four dimensions reads as

$$\begin{aligned}\partial_i \otimes \partial^{ij} + \partial^{ij} \otimes \partial_i = \partial_i\partial^{ij} =& 0\,,\\ \partial^{[ij} \otimes \partial^{jk]} = \partial^{[ij}\partial^{jk]} =& 0\,,\end{aligned} \tag{8.15}$$

Alternatively, the structure of (8.13) can be found using the flat derivatives

$$\mathcal{D}_{\bar{i}\bar{j}} \equiv E_{\bar{i}}{}^a E_{\bar{j}}{}^b \partial_{ab}\,. \tag{8.16}$$

To unflatten them, we have to use the vielbeins $e_i{}^{\bar{i}}$ again which results in

$$\begin{aligned}\hat{\partial}_i &= |e|^{-4/5} e_i{}^{\bar{i}} E_{\bar{i}}{}^a \partial_a = \partial_i\\ \hat{\partial}^{ij} &= \frac{1}{2}|e|^{-4/5} e_{\bar{i}}{}^i e_{\bar{j}}{}^j \epsilon^{\bar{i}\bar{j}\bar{k}\bar{l}} E_{\bar{k}}{}^a E_{\bar{l}}{}^b \partial_{ab}\,.\end{aligned} \tag{8.17}$$

These formulae are general and can be used in higher dimensions. Additionally, they give us a good double check for the vielbein. Note that the prefactor depends on the convention and cancels the prefactors from the vielbein.

After finding the right ingredients we can move on to find the right fluxes. Therefore, we have to decompose the fundamental group of the the embedding tensor of maximal gauged supergravities [89] under SL(5) $\longrightarrow$ SL(4) $\times \mathbb{R}^+$. The fundamental group is $\overline{\mathbf{15}} \oplus \overline{\mathbf{40}} \oplus \overline{\mathbf{10}}$ of SL(5) and the decomposition is given by

$$\begin{aligned}
\overline{\mathbf{15}} &\longrightarrow \overline{\mathbf{10}}_2 \oplus \overline{\mathbf{4}}_{-3} \oplus \mathbf{1}_{-8}\,,\\
\overline{\mathbf{40}} &\longrightarrow \overline{\mathbf{20}}_{-3} \oplus \mathbf{10}_2 \oplus \mathbf{6}_2 \oplus \overline{\mathbf{4}}_7\,,\\
\overline{\mathbf{10}} &\longrightarrow \overline{\mathbf{4}}_{-3} \oplus \mathbf{6}_2\,.
\end{aligned} \tag{8.18}$$

This strategy is similar to the encoding of the different fields, each of the components on the right hand side is an existing flux, and we have to find the correct combinations of the fields to construct them. As a general guideline, we have two hints from the decomposition how to find the correct form:

- We can use the $\mathbb{R}^+$-charge, which points towards possible combinations between the different fields and the different derivatives.
- We can use the representation itself to gain an idea on the degrees of freedom of the flux. For instance, a $\mathbf{1}$ denotes a scalar, so no free indices, a $\mathbf{4}$ denotes in four dimensions a vector (one free index), etc. Note that we have to distinguish between symmetric and anti-symmetric parts of the tensors for mixed symmetry fluxes. Additionally, for most mixed tensors, we will find a trace and a trace-free part which are different elements in the decomposition.

Having these ideas in mind, we can start with the most geometric flux, $\mathbf{1}_{-8}$. The only (simple) combination to reach such a low $\mathbb{R}^+$-charge, is $G_{ijkl} = 4\partial_{[i}C_{jkl]}$, which corresponds to the four-form flux in M-theory.

The next fluxes to look at arises from the elements with $\mathbb{R}^+$-charge -3, i.e. $\overline{\mathbf{20}}_{-3}$ and $2 \cdot \overline{\mathbf{4}}_{-3}$. Looking at the R^+-charges, we can see that the fluxes should depend on the spacetime derivative ∂_i plus an element with no $\mathbb{R}^+$-charge. This leads to two different possible fluxes.

- The first one is the spacetime derivative of the determinant of the seven-dimensional metric of the uncompactified dimensions, i.e. $\partial_i \Delta$, [88].
- The remaining two fluxes are the trace and the trace-free part of the geometric flux known from double field theory which combines the spacetime derivative acting on the vielbein

$$T_{ij}{}^{k} = 2e_{\bar{i}}{}^{k}\partial_{[i}e^{\bar{i}}{}_{j]} \in \overline{\mathbf{4}}_{-3} \oplus \overline{\mathbf{20}}_{-3}\,. \tag{8.19}$$

The geometric flux can also be expressed in terms of connections which can be introduced for both derivatives, defined as

$$\begin{aligned} \Gamma_{ij}{}^{k} &= e_{\bar{i}}{}^{k} \partial_i e^{\bar{i}}{}_{j} \,, \\ \hat{\Gamma}^{ij}{}_{k}{}^{l} &= e_{\bar{i}}{}^{l} \hat{\partial}^{ij} e^{\bar{i}}{}_{k} \,. \end{aligned} \tag{8.20}$$

Under spacetime diffeomorphisms, they transform as

$$\begin{aligned} \delta_\xi \Gamma_{ij}{}^{k} &= L_\xi \Gamma_{ij}{}^{k} + \partial_i \partial_j \xi^k \,, \\ \delta_\xi \hat{\Gamma}^{ij}{}_{k}{}^{l} &= L_\xi \hat{\Gamma}^{ij}{}_{k}{}^{l} + \hat{\partial}^{ij} \partial_k \xi^l \,. \end{aligned} \tag{8.21}$$

Note, that using these connections the geometric flux can be expressed as $T_{ij}{}^{k} = \Gamma_{[ij]}{}^{k}$.

The connections (8.20) prove to be useful when continuing with the identification of new arising fluxes. Taking the trace of $\hat{\Gamma}^{ij}{}_{k}{}^{l}$ the difference between ordinary Lie-derivative and spacetime diffeomorphism vanish due to the section condition. The resulting flux is known as τ-flux

$$\tau^{i,j} = \hat{\Gamma}^{ik}{}_{k}{}^{j} \,. \tag{8.22}$$

Here, the symmetric and anti-symmetric part are the representation of $\mathbf{6}_2 \oplus \mathbf{10}_2$.
Now, only three representations are left: $\mathbf{6}_2$, $\overline{\mathbf{10}}_2$ and $\mathbf{4}_7$. Let us check whether these are related to the non-geometric three form. By combining Ω^{ijk} and ∂_i we see that we reach a $\mathbb{R}^+$-charge equal to $+2$, which is exactly the correct charge. Computing the difference between spacetime diffeomorphism and Lie-derivative we get

$$\begin{aligned} \delta_\xi \left(\partial_i \Omega^{jkl} \right) =& L_\xi \left(\partial_i \Omega^{jkl} \right) - 3 \partial_i \partial^{[jk} \xi^{l]} - \Omega^{p[kl} \partial_p \partial_i \xi^{j]} \\ =& L_\xi \left(\partial_i \Omega^{jkl} \right) - 3 \hat{\partial}^{[jk} \partial_i \xi^{l]} \,. \end{aligned} \tag{8.23}$$

which corresponds to the remaining parts of $\hat{\Gamma}^{[jk}{}_{i}{}^{l]}$. Therefore, we find as flux

$$Q_i{}^{jkl} = \partial_i \Omega^{jkl} + 3 \hat{\Gamma}^{[jk}{}_{i}{}^{l]} \,. \tag{8.24}$$

This flux is known as “globally non-geometric flux” or as “Q-flux”. Splitting the flux into trace and trace-free part we find as corresponding representation

$$Q_i{}^{jkl} \in \mathbf{6}_2 \oplus \overline{\mathbf{10}}_2 \,. \tag{8.25}$$

The remaining component of the representation is $\mathbf{4}_7$. The $\mathbb{R}^+$-charge corresponds to a combination between $\hat{\partial}^{ij}$ and Ω^{ijk}. Due to the representation $\mathbf{4}$ it is expected that we will only have one free index, and have to anti-symmetrize over the remaining four as follows

$$\begin{aligned} \delta_\xi \left(\hat{\partial}^{i[j} \Omega^{klm]} \right) &= L_\xi \left(\hat{\partial}^{i[j} \Omega^{klm]} \right) - 3 \Omega^{p[lm} \hat{\partial}^{|i|j} \partial_p \xi^{k]} - 3 \hat{\partial}^{i[j} \partial^{kl} \xi^{m]} \\ &= L_\xi \left(\hat{\partial}^{i[j} \Omega^{klm]} \right) - 3 \Omega^{p[lm} \partial^{|i|j} \partial_p \xi^{k]} - 3 \Omega^{pi[j} \partial_p \partial^{kl} \xi^{m]} \\ &\qquad + 3 \Omega^{p[lm} \Omega^{j|io|} \partial_o \partial_p \xi^{k]} - 3 \partial^{i[j} \partial^{kl} \xi^{m]} \,. \end{aligned} \tag{8.26}$$

To show that this vanishes we can follow a fixed set of rules to check whether all terms on the right side vanish:

- Anti-symmetrizing over $d+1$ indices in d dimensions always leads to a vanishing quantity, since there will be always a pair of indices having the same index value.
- Using the section condition whenever possible.
- Summing over indices with an odd and an even part is also a typical trick to show that certain terms vanish.
- In most cases it is necessary to split the improved derivatives. Thus, they only simplify the notation of the fluxes but not necessarily the computations itself.

For clarity, let us consider a short example. Let us take the term $\partial^{i[j}\partial^{kl}\xi^{m]}$ and anti-symmetrize in all of its indices as follows

$$0 = \partial^{[ij}\partial^{kl}\xi^{m]} = \frac{1}{5}\left(4\partial^{i[j}\partial^{kl}\xi^{m]} + \partial^{[jk}\partial^{lm]}\xi^{i]}\right) = \frac{4}{5}\partial^{i[j}\partial^{kl}\xi^{m]}\,. \tag{8.27}$$

The prefactor $\frac{1}{5}$ arises from the used convention. It can be shown that the other terms in equation (8.26) vanish in a similar fashion.[1]

As a result, we find for the representation $\mathbf{4}_7$ the locally non-geometric flux (also known as R-flux):

$$R^{i,jklm} = 4\hat{\partial}^{i[j}\Omega^{klm]} \in \mathbf{4}_7\,. \tag{8.28}$$

This R-flux appears in the duality chain in section 7.3.2 as well and in string theory it corresponds to the known NS-NS R-flux, provided that we reduce a suitable dimension on a torus. Knowing the decomposition of the fundamental group we can ensure that we found all fluxes in four dimensions. After reviewing the results of [88] (and showing the key parts of the computations) we can move on to the cases where $D = 5, 6, 7$.

8.1.2 Five-dimensional backgrounds

We can now continue to use this framework to generalize the results to higher dimensions. We start with the six-dimensional SUGRA which corresponds to the SO(5, 5) exceptional field theory in five dimensions, decompose its fundamental representation $\overline{\mathbf{16}}$ under SO(5, 5) ⟶ SL(5) × $\mathbb{R}^+$ as

$$\overline{\mathbf{16}} \longrightarrow \mathbf{10}_1 \oplus \overline{\mathbf{5}}_{-3} \oplus \mathbf{1}_5\,. \tag{8.29}$$

We can find directly the right notation of the derivatives, given by

$$\partial_i \in \overline{\mathbf{5}}_{-3}\,, \qquad \partial^{ij} \in \mathbf{10}_1\,, \qquad \partial^z \in \mathbf{1}_5\,. \tag{8.30}$$

Looking at the adjoint decomposition we find

$$\mathbf{45} \longrightarrow \mathbf{24}_0 \oplus \mathbf{10}_{-4} \oplus \overline{\mathbf{10}}_4 \oplus \mathbf{1}_0\,, \tag{8.31}$$

---

[1] In low dimensions such computation are quite simple, whereas in higher dimensions (specially for $d = 7$) more and more terms have to be taken into account.

and therefore, we can identify our fields as

$$e_{\bar{i}}{}^{i} \in \mathbf{24}_0 \oplus \mathbf{1}_0\,, \qquad C_{ijk} \in \mathbf{10}_{-4}\,, \qquad \Omega^{ijk} \in \overline{\mathbf{10}}_4\,. \tag{8.32}$$

Knowing the field content, we can construct the vielbein and compute the transformation behavior of the fields as well as defining the improved derivatives. In five-dimensional exceptional field theory the vielbein in the non-geometric frame is defined as

$$E_{\bar{M}}{}^{M} \equiv \begin{pmatrix} |e|^{-1/4} e_{\bar{i}}{}^{i} & 0 & 0 \\ |e|^{-1/4}\Omega^{\bar{i}_1\bar{i}_2 i} & |e|^{-1/4} e^{\bar{i}_1\bar{i}_2}_{i_1 i_2} & 0 \\ -\frac{1}{12}|e|^{-1/4}\epsilon_{k_1k_2k_3k_4k_5}\Omega^{ik_1k_2}\Omega^{k_3k_4k_5} & -\frac{1}{6}|e|^{-1/4}\epsilon_{i_1i_2k_1k_2k_3}\Omega^{k_1k_2k_3} & |e|^{3/4} \end{pmatrix}\,, \tag{8.33}$$

where

$$e^{\bar{i}\bar{j}}_{ij} = e^{[\bar{i}}{}_{i} e^{\bar{j}]}{}_{j}\,. \tag{8.34}$$

Computing the generalized Lie-derivative of the generalized vielbein with a spacetime vector $V^M = (\xi^i,\, 0,\, 0)$ we find the same transformation behavior as in the four-dimensional exceptional field theory, i.e.

$$\begin{aligned} \delta_\xi e_{\bar{i}}{}^{i} &= L_\xi e_{\bar{i}}{}^{i}\,. \\ \delta_\xi C_{ijk} &= L_\xi C_{ijk}\,. \\ \delta_\xi \Omega^{ijk} &= L_\xi \Omega^{ijk} - 3\partial^{[ij}\xi^{k]}\,. \end{aligned} \tag{8.35}$$

As in the four-dimensional exceptional field theory, we state the section conditions which is given by

$$\begin{gathered} \partial_i \otimes \partial^{ij} + \partial^{ij} \otimes \partial_i = \partial_i \partial^{ij} = 0\,, \\ \partial_i \otimes \partial^z + \partial^z \otimes \partial_i + \frac{1}{2}\epsilon_{ijklm}\partial^{jk} \otimes \partial^{lm} = \partial_i\partial^z f + \frac{1}{4}\epsilon_{ijklm}\partial^{jk}\partial^{lm} f = 0\,. \end{gathered} \tag{8.36}$$

Note that the second section condition changed, while the first one stayed the same. We will have to put special attention whenever we use the second section condition. Having defined the section condition, we can now find the improved derivatives which are defined as

$$\begin{aligned} \hat{\partial}_i &= |e|^{1/4} e^{\bar{i}}{}_{i} E_{bi}\bar{i}\, M \partial_M \\ &= \partial_i\,, \\ \hat{\partial}^{ij} &= |e|^{1/4} e_{\bar{i}}{}^{i} e_{\bar{j}}{}^{j} E^{\bar{i}\bar{j}\,M}\partial_M \\ &= \partial^{ij} + \Omega^{ijk}\partial_k\,, \\ \hat{\partial}^z &= |e|^{1/4} E^{\bar{z}\,M}\partial_M \\ &= |e|\partial^z - \frac{1}{6}\epsilon_{ijklm}\Omega^{ijk}\partial^{lm} - \frac{1}{12}\epsilon_{jklmn}\Omega^{ijk}\Omega^{lmn}\partial_i\,, \end{aligned} \tag{8.37}$$

and satisfy

$$\delta_\xi \hat{\partial}_j \phi = L_\xi \hat{\partial}_j \phi\,, \qquad \delta_\xi \hat{\partial}^{ij}\phi = L_\xi \hat{\partial}^{ij}\phi\,, \qquad \delta_\xi \hat{\partial}^z \phi = L_\xi \hat{\partial}^z \phi = \xi^i \partial_i \hat{\partial}^z \phi\,. \tag{8.38}$$

Knowing the components of the fluxes, we can decompose the fundamental group $\overline{\mathbf{16}} \oplus \overline{\mathbf{144}}$ of $\mathrm{SO}(5,5)$ under $\mathrm{SL}(5) \times \mathbb{R}^+ \subset \mathrm{SO}(5,5)$ and find

$$\begin{aligned} \overline{\mathbf{16}} &\longrightarrow \mathbf{10}_1 \oplus \overline{\mathbf{5}}_{-3} \oplus \mathbf{1}_5 \,, \\ \overline{\mathbf{144}} &\longrightarrow \mathbf{15}_1 \oplus \overline{\mathbf{45}}_{-3} \oplus \mathbf{24}_5 \oplus \mathbf{5}_{-7} \oplus \mathbf{40}_1 \oplus \mathbf{10}_1 \oplus \overline{\mathbf{5}}_{-3} \,. \end{aligned} \tag{8.39}$$

To simplify the notation of the fluxes, we introduce again the connections for the different derivatives

$$\begin{aligned} \Gamma_{ij}{}^k &= e_{\bar{i}}{}^k \partial_i e^{\bar{i}}{}_j \,, \\ \hat{\Gamma}^{ij}{}_k{}^l &= e_{\bar{i}}{}^l \hat{\partial}^{ij} e^{\bar{i}}{}_k \,, \\ \hat{\Gamma}^z{}_i{}^j &= e_{\bar{i}}{}^i \hat{\partial}^z e^{\bar{i}}{}_j \,. \end{aligned} \tag{8.40}$$

These objects transform under spacetime diffeomorphisms as in the SL(5) exceptional field theory,

$$\begin{aligned} \delta_\xi \Gamma_{ij}{}^k &= L_\xi \Gamma_{ij}{}^k + \partial_i \partial_j \xi^k \,, \\ \delta_\xi \hat{\Gamma}^{ij}{}_k{}^l &= L_\xi \hat{\Gamma}^{ij}{}_k{}^l + \hat{\partial}^{ij} \partial_k \xi^l \,, \\ \delta_\xi \hat{\Gamma}^z{}_i{}^j &= L_\xi \hat{\Gamma}^z{}_i{}^j + \hat{\partial}^z \partial_i \xi^j \,. \end{aligned} \tag{8.41}$$

Comparing the five-dimensional case to the four-dimensional case, we realize that most components and the fluxes stay the same (except for the second section condition) and therefore, we can use the results from the four-dimensional case for most of the fluxes. The list is given as

$$\begin{aligned} Q_i{}^{jkl} &= \partial_i \Omega^{jkl} + 3 \hat{\Gamma}^{[jk}{}_i{}^{l]} \in \mathbf{10}_1 \oplus \mathbf{40}_1 \,, \\ \tau^{i,j} &= \hat{\Gamma}^{ip}{}_p{}^j \in \mathbf{10}_1 \oplus \mathbf{15}_1 \,, \\ G_{ijkl} &= 4 \partial_{[i} C_{jkl]} \in \mathbf{5}_{-7} \,, \\ \partial_i \Delta &\in \overline{\mathbf{5}}_{-3} \,, \\ T_{ij}{}^k &= \Gamma_{[ij]}{}^k \in \overline{\mathbf{5}}_{-3} \oplus \overline{\mathbf{45}}_{-3} \,. \end{aligned} \tag{8.42}$$

Therefore, when we change the dimension from four to five in exceptional field theory, the *more geometric* fluxes stay the same and will reappear in higher dimensions. Therefore, we focus on the non-geometric fluxes in higher dimensions, where we cannot simply uplift the locally non-geometric R-flux due to the change in the section condition. After an involved computation, we find the following combination for the flux

$$R^{i,jklm} = 4 \hat{\partial}^{i[j} \Omega^{klm]} - 2 e_{\bar{i}}{}^{[j} \epsilon^{klm]in} \hat{\partial}^z e^{\bar{i}}{}_n = 4 \hat{\partial}^{i[j} \Omega^{klm]} + 2 \epsilon^{in[jkl} \hat{\Gamma}^z{}_n{}^{l]} \,. \tag{8.43}$$

This flux involves also the new dual derivative which corresponds to the wrapping coordinates of the M5-brane. As a consequence of the five-dimensional compactification, the new R-flux contains a totally anti-symmetric part, which has to be viewed as a different R-flux. This flux can be written as follows

$$R^{ijklm} \equiv 5 R^{[i,jklm]} = 20 \hat{\partial}^{[ij} \Omega^{klm]} - 2 \epsilon^{ijklm} \hat{\partial}^z |e| \,. \tag{8.44}$$

Comparing the degrees of freedom of the locally non-geometric R-fluxes R^{ijklm} and $R^{i,jklm}$, we see that the flux corresponds to different parts of the embedding tensor: While the new total anti-symmetric R-flux R^{ijklm} represents the $\mathbf{1}_5 \in \overline{\mathbf{16}}$, the R-flux with mixed symmetry structure $R^{i,jklm} - R^{[i,jklm]}$ belongs to the $\mathbf{24}_5 \in \mathbf{144}$. Comparing this knowledge with the reduction from M-theory to string theory in section 7.3.2, we can see the consequences. While the $R^{i,jklm}$ reduces to the NS-NS R-flux in string theory when compactifying in appropriate directions, the new R-flux R^{ijklm} will necessarily involve the R-R sector of string theory.

The symmetry group decomposition of the embedding tensor assures us that we have found all existing non-geometric fluxes. In the next section we discuss the case for $d = 6$.

8.1.3 Six-dimensional backgrounds

Let us now apply the same strategy for the case with six compactified dimensions. We start with the fundamental representation of five-dimensional SUGRA and decompose the $\overline{\mathbf{27}}$ and the adjoint $\mathbf{78}$ under $\mathrm{SL}(6) \times \mathbb{R}^+ \subset \mathrm{E}_{6(6)}$

$$\begin{aligned} \overline{\mathbf{27}} &\longrightarrow \mathbf{15}_0 \oplus \overline{\mathbf{6}}_2 \oplus \overline{\mathbf{6}}_{-2}\,, \\ \mathbf{78} &\longrightarrow \mathbf{35}_0 \oplus \mathbf{20}_2 \oplus \mathbf{20}_{-2} \oplus \mathbf{1}_4 \oplus \mathbf{1}_0 \oplus \mathbf{1}_{-4}\,. \end{aligned} \tag{8.45}$$

Using these decompositions we find for the derivatives the structure

$$\partial_i \in \overline{\mathbf{6}}_{-2}\,, \qquad \partial^{ij} \in \mathbf{15}_0\,, \qquad \partial^{ijklm} \in \overline{\mathbf{6}}_2\,, \tag{8.46}$$

which is the same as in SO(5, 5) EFT – we only switched the convention to not get confused with a second derivative with only one lower index. The possible fields in $\mathrm{E}_{6(6)}$ are

$$\begin{aligned} e_{\bar{i}}{}^{i} \in \mathbf{35}_0 \oplus \mathbf{1}_0\,, \qquad C_{ijk} &\in \mathbf{20}_{-4}\,, \qquad C_{ijklmn} \in \mathbf{1}_{-4}\,, \\ \Omega^{ijk} &\in \overline{\mathbf{20}}_4\,, \qquad \Omega^{ijklmn} \in \overline{\mathbf{1}}_4\,. \end{aligned} \tag{8.47}$$

As we can see, we have two additional new fields, namely, the geometric 6-form C_{ijklmn} and the non-geometric 6-form Ω^{ijklmn}. They appear because we now have six compact dimensions and therefore all six legs of the six-form can lie in compactified directions. As a consequence we can get additional fluxes.

Following the convention, we get for the non-geometric parameterization of the vielbein

$$E_{\bar{M}}{}^{M} = \begin{pmatrix} |e|^{-1/3} e_{\bar{i}}{}^{i} & 0 & 0 \\ |e|^{-1/3} \Omega^{\bar{i}_1 \bar{i}_2 i} & |e|^{-1/3} e^{\bar{i}_1 \bar{i}_2}_{i_1 i_2} & 0 \\ \sqrt{\frac{5}{6}} |e|^{-1/3} \left(\Omega^{\bar{i}_1 \bar{i}_2 \bar{i}_3 \bar{i}_4 \bar{i}_5 i} + \Omega^{[\bar{i}_1 \bar{i}_2 \bar{i}_3} \Omega^{\bar{i}_4 \bar{i}_5] i} \right) & \sqrt{\frac{10}{3}} |e|^{-1/3} e^{[\bar{i}_1 \bar{i}_2}_{i_1 i_2} \Omega^{\bar{i}_3 \bar{i}_4 \bar{i}_5]} & |e|^{-1/3} e^{\bar{i}_1 \bar{i}_2 \bar{i}_3 \bar{i}_4 \bar{i}_5}_{i_1 i_2 i_3 i_4 i_5} \end{pmatrix}, \tag{8.48}$$

where

$$e^{\bar{i}_1 \bar{i}_2 \bar{i}_3 \bar{i}_4 \bar{i}_5}_{i_1 i_2 i_3 i_4 i_5} = e^{[\bar{i}_1}{}_{i_1} e^{\bar{i}_2}{}_{i_2} e^{\bar{i}_3}{}_{i_3} e^{\bar{i}_4}{}_{i_4} e^{\bar{i}_5]}{}_{i_5}\,. \tag{8.49}$$

For the transformation behavior under spacetime diffeomorphisms of the different fields, we find that

$$\begin{aligned}
\delta_\xi e_{\bar{i}}{}^i &= L_\xi e_{\bar{i}}{}^i\,,\\
\delta_\xi C_{ijk} &= L_\xi C_{ijk}\,,\\
\delta_\xi C_{ijklmn} &= L_\xi C_{ijk}\,,\\
\delta_\xi \Omega^{ijk} &= L_\xi \Omega^{ijk} - 3\partial^{[ij}\xi^{k]}\,,\\
\delta_\xi \Omega^{ijklmn} &= L_\xi \Omega^{ijklmn} + \frac{3}{5}\partial^{[ijklm}\xi^{n]} - 6\Omega^{[ijk}\partial^{lm}\xi^{n]}\,.
\end{aligned} \tag{8.50}$$

As before, let us introduce the improved derivative operators

$$\begin{aligned}
\hat{\partial}_i &= |e|^{1/3} e^{\bar{i}}{}_i E_{bi}{}^{\bar{i}}\, M \partial_M\\
&= \partial_i\,,\\
\hat{\partial}^{ij} &= |e|^{1/3} e_{\bar{i}}{}^i e_{\bar{j}}{}^i E^{\bar{i}\bar{j}\,M}\partial_M = \partial^{ij} + \Omega^{ijk}\partial_k\,,\\
\hat{\partial}^{ijklm} &= -\sqrt{5!}\,|e|^{1/3} e_{\bar{i}}{}^i e_{\bar{j}}{}^j e_{\bar{k}}{}^k e_{\bar{l}}{}^l e_{\bar{m}}{}^m E^{\bar{i}\bar{j}\bar{k}\bar{l}\bar{m}\,M}\partial_M\\
&= \partial^{ijklm} - 20\Omega^{[ijk}\partial^{lm]} - 10\left(\Omega^{ijklmn} + \Omega^{[ijk}\Omega^{lm]n}\right)\partial_n\,,
\end{aligned} \tag{8.51}$$

where

$$\partial_M = \left(\partial_i,\, \partial^{ij},\, -\frac{1}{\sqrt{5!}}\partial^{ijklm}\right)\,, \tag{8.52}$$

so that $\hat{\partial}_i\phi$ $\hat{\partial}^{ij}\phi$ and $\hat{\partial}^{ijklm}\phi$ are spacetime tensors, where ϕ is a scalar. Additionally, we state the section conditions which are

$$\begin{gathered}
\partial_i \otimes \partial^{ij} + \partial^{ij} \otimes \partial_i g = \partial_i\partial^{ij} = 0\,,\\
\partial^{[ij} \otimes \partial^{kl]} + \frac{1}{12}\left(\partial_m \otimes \partial^{ijklm} + \partial^{ijklm} \otimes \partial_m\right) = \partial^{[ij}\partial^{kl]}f - \frac{1}{6}\partial_m\partial^{ijklm}f = 0\,,\\
\partial^{i[j} \otimes \partial^{klmnp]} - \partial^{[klmnp} \otimes \partial^{j]i} = \partial^{i[j}\partial^{klmnp]}f = 0\,.
\end{gathered} \tag{8.53}$$

Before we start with the decomposition and the search for the non-geometric fluxes, we state the notion of the connection-like objects

$$\begin{aligned}
\Gamma_{ij}{}^k &= e_{\bar{i}}{}^k \partial_i e^{\bar{i}}{}_j\,,\\
\hat{\Gamma}^{ij}{}_k{}^l &= e_{\bar{i}}{}^l \hat{\partial}^{ij} e^{\bar{i}}{}_k\,,\\
\hat{\Gamma}^{ijklm}{}_n{}^p &= e_{\bar{i}}{}^p \hat{\partial}^{ijklm} e^{\bar{i}}{}_n\,.
\end{aligned} \tag{8.54}$$

Their transformation behavior under spacetime diffeomorphisms can be found in section 8.1.2.

To ensure that we later find all possible fluxes we have to decompose the symmetry group of the $\mathrm{E}_{6(6)}$ embedding tensor under $\mathrm{SL}(6) \times \mathbb{R}^+ \subset \mathrm{E}_{6(6)}$ which is given as

$$\begin{aligned}
\overline{\mathbf{27}} &\to \mathbf{15}_0 \oplus \overline{\mathbf{6}}_2 \oplus \overline{\mathbf{6}}_{-2}\,,\\
\overline{\mathbf{351}} &\to \mathbf{21}_0 \oplus \overline{\mathbf{84}}_2 \oplus \overline{\mathbf{84}}_{-2} \oplus \mathbf{105}_0 \oplus \mathbf{15}_4 \oplus \mathbf{15}_0 \oplus \mathbf{15}_{-4} \oplus \overline{\mathbf{6}}_2 \oplus \overline{\mathbf{6}}_{-2}\,.
\end{aligned} \tag{8.55}$$

First, we start with finding the geometric fluxes which are

$$\begin{aligned} G_{ijkl} &= 4\partial_{[i}C_{jkl]} \in \mathbf{15}_{-4}\,, \\ \partial_i\Delta &\in \overline{\mathbf{6}}_{-2}\,, \\ T_{ij}{}^k &= \Gamma_{[ij]}{}^k \in \overline{\mathbf{6}}_{-2} \oplus \overline{\mathbf{84}}_{-2}\,, \\ \tau^{i,j} &= \hat{\Gamma}^{ip}{}_p{}^j \in \mathbf{15}_0 \oplus \mathbf{21}_0\,. \end{aligned} \tag{8.56}$$

These fluxes are the same as in four and five dimensions. Note that the flux $\tau^{i,j}$ technically is a non-geometric flux due to the dual derivative $\hat{\partial}^{ij}$. Additionally, we find the Q-flux associated with Ω^{ijk} in six dimensions

$$Q_i{}^{jkl} = \partial_i\Omega^{jkl} + 3\hat{\Gamma}^{[jk}{}_i{}^{l]} \in \mathbf{15}_0 \oplus \mathbf{105}_0\,. \tag{8.57}$$

Due to the existence of the non-geometric 6-form Ω^{ijklmn}, we can find another Q-flux in six dimensions as

$$Q_i{}^{jklmnp} = \partial_i\Omega^{jklmnp} + 2\Omega^{[jkl}\partial_i\Omega^{mnp]} - \frac{3}{5}\hat{\Gamma}^{[jklmn}{}_i{}^{p]} \in \overline{\mathbf{6}}_2\,. \tag{8.58}$$

Therefore, we are left with the “most non-geometric” representations of $\mathrm{E}_{6(6)} \to \mathrm{SL}(6)\times\mathbb{R}^+$, which are: $\overline{\mathbf{6}}_2 \oplus \overline{\mathbf{84}}_2 \oplus \in \mathbf{15}_4$.

The representation $\overline{\mathbf{6}}_2 \oplus \overline{\mathbf{84}}_2$ belongs again to the local non-geometric R-flux known from four and five dimensions,

$$R^{i,jklm} = 4\hat{\partial}^{i[j}\Omega^{klm]} + 2\hat{\Gamma}^{in[jkl}{}_n{}^{m]}\,, \tag{8.59}$$

with the known split in totally anti-symmetric and mixed symmetric part of the flux.

Therefore, we are left to find the final $\mathbf{15}_4 \subset \mathbf{351}$ in the embedding tensor. The $\mathbb{R}^+$ can be constructed combining using $\hat{\partial}^{ijklm}$ on Ω^{ijk} or $\hat{\partial}^{ij}$ on Ω^{ijklmn}. To construct this new R-flux we have to combine both possibilities. Then, the new flux is given by

$$R^{ij} = \frac{1}{5!}\epsilon_{klmnpq}\hat{\partial}^{klmnp}\Omega^{qij} + \frac{1}{72}\epsilon_{klmnpq}\hat{\partial}^{ij}\Omega^{klmnpq} + \frac{1}{36}\epsilon_{klmnpq}\Omega^{klm}\hat{\partial}^{ij}\Omega^{npq}\,. \tag{8.60}$$

One can notice that in all fluxes with an derivative acting on a non-geometric six-form $\partial^M\Omega^{ijklmn}$ we have to introduce a term with two non-geometric three-forms $\Omega^{ijk}\hat{\partial}^M\Omega^{lmn}$.

8.1.4 Seven-dimensional backgrounds

Finally, we are going to apply this strategy to seven compactified dimensions. First, we start decomposing the fundamental group **56** and the adjoint **133** under $\mathrm{E}_{7(7)} \to \mathrm{SL}(8) \to \mathrm{SL}(7) \times \mathbb{R}^+$

$$\begin{aligned} \mathbf{56} &\longrightarrow \overline{\mathbf{7}}_{-6} \oplus \mathbf{21}_{-2} \oplus \overline{\mathbf{21}}_2 \oplus \mathbf{7}_6\,, \\ \mathbf{133} &\longrightarrow \mathbf{35}_{-4} \oplus \overline{\mathbf{7}}_{-8} \oplus \mathbf{48}_0 \oplus \mathbf{1}_0 \oplus \mathbf{35}_4 \oplus \overline{\mathbf{7}}_8\,. \end{aligned} \tag{8.61}$$

Again, we find the structure of the derivatives to be

$$\partial_i \in \overline{\mathbf{7}}_{-6}\,, \qquad \partial^{ij} \in \mathbf{21}_{-2}\,, \qquad \partial_{ij} \in \overline{\mathbf{21}}_{2}\,, \qquad \partial^i \in \mathbf{7}_{6}\,, \tag{8.62}$$

as well as the existing fields

$$\begin{aligned} e_{\bar{i}}{}^{i} \in \mathbf{48}_0 \oplus \mathbf{1}_0\,, \qquad & C_{ijk} \in \mathbf{35}_{-4}\,, \qquad && C_{ijklmn} \in \mathbf{7}_{-4}\,, \\ & \Omega^{ijk} \in \overline{\mathbf{35}}_{4}\,, && \Omega^{ijklmn} \in \overline{\mathbf{7}}_{4}\,. \end{aligned} \tag{8.63}$$

As next step we introduce the generalized vielbein. To shorten our notation, we introduce dualized 3-forms and 6-forms as

$$\begin{aligned} V_{ijkl} &= \epsilon_{ijklmnp}\Omega^{mnp}\,, \\ W_{ij}{}^{k} &= V_{ijlm}\Omega^{klm}\,, \\ X_{ij}{}^{k} &= \epsilon_{ijlmnpq}\Omega^{lmnpqk}\,. \end{aligned} \tag{8.64}$$

Using them, the generalized vielbein of $\mathrm{E}_{7(7)}$ in the non-geometric parameterization is given by

$$E_{\bar{M}}{}^{M} = \begin{pmatrix} |e|^{-1/2}e_{\bar{i}}{}^{i} & 0 & 0 & 0 \\ |e|^{-1/2}\Omega^{\bar{i}_1\bar{i}_2 i} & |e|^{-1/2}e^{\bar{i}_1\bar{i}_2}_{i_1 i_2} & 0 & 0 \\ \frac{1}{24}|e|^{-1/2}\left(X_{\bar{i}_1\bar{i}_2}{}^{i} + W_{\bar{i}_1\bar{i}_2}{}^{i}\right) & \frac{1}{12}|e|^{-1/2}V_{\bar{i}_1\bar{i}_2 i_1 i_2} & |e|^{1/2}e^{i_1 i_2}_{\bar{i}_1\bar{i}_2} & 0 \\ |e|^{-1/2}\frac{1}{144}\left(W_{jk}{}^{i}\Omega^{\bar{i}jk} - 3X_{jk}{}^{i}\Omega^{\bar{i}jk}\right) & \frac{1}{48}|e|^{-1/2}\left(W_{i_1 i_2}{}^{\bar{i}} - X_{i_1 i_2}{}^{\bar{i}}\right) & \frac{1}{2}|e|^{1/2}\Omega^{\bar{i}i_1 i_2} & |e|^{1/2}e^{\bar{i}}{}_{i} \end{pmatrix}. \tag{8.65}$$

From the action of the generalized Lie derivative on the generalized vielbein we can deduce the transformation laws of the 3-form and the 6-form under a spacetime diffeomorphism with the vector field $V^M = (\xi^i,\, 0,\, 0,\, 0)$,

$$\begin{aligned} \delta_\xi e_{\bar{i}}{}^{i} &= L_\xi e_{\bar{i}}{}^{i}\,. \\ \delta_\xi C_{ijk} &= L_\xi C_{ijk}\,. \\ \delta_\xi C_{ijklmn} &= L_\xi C_{ijk}\,. \\ \delta_\xi \Omega^{ijk} &= L_\xi \Omega^{ijk} - 3\partial^{[ij}\xi^{k]}\,, \\ \delta_\xi \Omega^{ijklmn} &= L_\xi \Omega^{ijklmn} - \frac{1}{10}\epsilon^{ijklmnp}\partial_{pq}\xi^{q} - 6\Omega^{[ijk}\partial^{lm}\xi^{n]}\,. \end{aligned} \tag{8.66}$$

Note that the different prefactors between the six and seven-dimensional case arise from the different conventions of the dual derivative corresponding to the M5-brane.

Again, we are using the generalized vielbein to find improved derivatives which fulfill the conditions

$$\delta_\xi \hat{\partial}_i\phi = L_\xi \hat{\partial}_i\phi\,, \qquad \delta_\xi \hat{\partial}^{ij}\phi = L_\xi \hat{\partial}^{ij}\phi\,, \qquad \delta_\xi \hat{\partial}_{ij}\phi = L_\xi \hat{\partial}_{ij}\phi\,, \qquad \delta_\xi \hat{\partial}^{i}\phi = L_\xi \hat{\partial}^{i}\phi\,. \tag{8.67}$$

In seven dimensions those derivatives are

$$\begin{aligned}
\hat{\partial}^{ij} &= |e|^{1/2} e_{\bar{i}}{}^{i} e_{\bar{j}}{}^{i} E^{\bar{i}\bar{j}\,M} \partial_M = \partial^{ij} + \Omega^{ijk}\partial_k \,, \\
\hat{\partial}_{ij} &= -2\,|e|^{1/2} e^{\bar{i}}{}_i e^{\bar{j}}{}_j E_{\bar{i}\bar{j}}{}^{M} \partial_M \\
&= |e|\partial_{ij} - \frac{1}{6}\epsilon_{ijklmnp}\Omega^{klm}\partial^{np} - \frac{1}{12}\epsilon_{ijklmnp}\left(\Omega^{klmnpq} + \Omega^{klm}\Omega^{npq}\right)\partial_q \,, \\
\hat{\partial}^{i} &= |e|^{1/2} e_{\bar{i}}{}^{i} E^{\bar{i}\,M}\partial_M \\
&= |e|\partial^i - \frac{1}{4}|e|\Omega^{ijk}\partial_{jk} + \frac{1}{48}\epsilon_{jklmnpq}\left(\Omega^{jkl}\Omega^{imn} - \Omega^{jklmni}\right)\partial^{pq} \\
&\quad + \frac{1}{144}\epsilon_{jklmnpq}\Omega^{pqr}\left(\Omega^{jkl}\Omega^{imn} - 3\Omega^{jklmni}\right)\partial_r \,,
\end{aligned} \tag{8.68}$$

where

$$\partial_M = \left(\partial_i,\, \partial^{ij},\, -\frac{1}{2}\partial_{ij},\, \partial^i\right) . \tag{8.69}$$

They fulfill the condition above up to the section conditions which are in seven dimensions given as

$$\begin{aligned}
\partial_i \otimes \partial^{ij} + \partial^{ij} \otimes \partial_i &= \partial_i\partial^{ij} = 0 \,, \\
\partial^i \otimes \partial_{ij} + \partial_{ij} \otimes \partial^{ij} &= \partial^i\partial_{ij} = 0 \,, \\
\partial_i \otimes \partial^i &= \partial_i\partial^i = 0 \,, \\
\partial^{ij} \otimes \partial_{ij} &= \partial_{ij}\partial^{ij} f = 0 \,, \\
\partial_i \otimes \partial^j + \partial^j \otimes \partial_i - \frac{1}{2}\partial_{ik} \otimes \partial^{jk} - \frac{1}{2}\partial^{jk} \otimes \partial_{ik} &= \partial_i\partial^j - \frac{1}{2}\partial_{ik}\partial^{jk} = 0 \,, \\
\partial_{[ij} f \partial_{kl]} g - \frac{1}{3}\epsilon_{ijklmnp}\left(\partial^{mn} f \partial^p g + \partial^p f \partial^{mn}\right) g &= \partial_{[ij}\partial_{kl]} f - \frac{2}{3}\epsilon_{ijklmnp}\partial^{mn}\partial^p = 0 \,, \\
\partial_{[ij} \otimes \partial_{k]} + \partial_{[i} \otimes \partial_{jk]} + \frac{1}{6}\epsilon_{ijklmnp}\partial^{mn} \otimes \partial^{np} &= \partial_{[ij}\partial_{k]} + \frac{1}{12}\epsilon_{ijklmnp}\partial^{mn}\partial^{np} = 0 \,.
\end{aligned} \tag{8.70}$$

As in the dimensions before, we first introduce the connection-like terms for ease of notation. They are given as

$$\begin{aligned}
\Gamma_{ij}{}^{k} &= e_{\bar{i}}{}^{k}\partial_i e^{\bar{i}}{}_j \,, \\
\hat{\Gamma}^{ij}{}_{k}{}^{l} &= e_{\bar{i}}{}^{l}\hat{\partial}^{ij} e^{\bar{i}}{}_k \,, \\
\hat{\Gamma}_{ijk}{}^{p} &= e_{\bar{i}}{}^{p}\hat{\partial}^{ij} e^{\bar{i}}{}_k \,, \\
\hat{\Gamma}^{i}{}_{j}{}^{l} &= e_{\bar{i}}{}^{l}\hat{\partial}^{i} e^{\bar{i}}{}_j \,.
\end{aligned} \tag{8.71}$$

As before, we start with the decomposition of the embedding tensor of four-dimensional maximal gauged SUGRA. Therefore, we have to decompose $\mathbf{56} \oplus \mathbf{912}$ of $\mathrm{E}_{7(7)}$ under $\mathrm{E}_{7(7)} \longrightarrow \mathrm{SL}(7) \times \mathbb{R}^+$, which is given by

$$\begin{aligned}
\mathbf{56} \longrightarrow & \overline{\mathbf{7}}_{-6} \oplus \mathbf{21}_{-2} \oplus \overline{\mathbf{21}}_{2} \oplus \mathbf{7}_{6} \,, \\
\mathbf{912} \longrightarrow & \mathbf{1}_{-14} \oplus \mathbf{35}_{-10} \oplus \overline{\mathbf{140}}_{-6} \oplus \overline{\mathbf{7}}_{-6} \oplus \mathbf{224}_{-2} \oplus \mathbf{21}_{-2} \oplus \mathbf{28}_{-2} \\
& \oplus \overline{\mathbf{28}}_{2} \oplus \overline{\mathbf{21}}_{2} \oplus \overline{\mathbf{224}}_{2} \oplus \mathbf{7}_{6} \oplus \mathbf{140}_{6} \oplus \overline{\mathbf{35}}_{10} \oplus \mathbf{1}_{14} \,.
\end{aligned} \tag{8.72}$$

As before, we start with the geometric fluxes

$$\begin{aligned} G_{ijkl} &= 4\partial_{[i}C_{jkl]} \in \mathbf{35}_{-10}\,, \\ G_{ijklmnp} &= 7\partial_{[i}C_{jklmnp} - 35C_{[ijk}G_{lmnp]} \in \mathbf{1}_{-14}\,, \\ \partial_i\Delta &\in \overline{\mathbf{7}}_{-6}\,, \\ T_{ij}{}^k &= \Gamma_{[ij]}{}^k \in \overline{\mathbf{7}}_{-6} \oplus \overline{\mathbf{140}}_{-6}\,, \end{aligned} \tag{8.73}$$

Here, the only new flux is $G_{ijklmnp}$, which is known from SUGRA. Furthermore, the Q-fluxes and the τ-fluxes are found to be seven-dimensional spacetime tensors

$$\begin{aligned} \tau^{i,j} &= \hat{\Gamma}^{ip}{}_p{}^j \in \mathbf{21}_{-2} \oplus \mathbf{28}_{-2}\,, \\ Q_i{}^{jkl} &= \partial_i\Omega^{jkl} + 3\hat{\Gamma}^{[jk}{}_i{}^{l]} \in \mathbf{21}_{-2} \oplus \mathbf{224}_{-2}\,, \\ Q_i{}^{jklmnp} &= \partial_i\Omega^{jklmnp} + 2\Omega^{[jkl}\partial_i\Omega^{mnp]} - \frac{3}{5}\hat{\Gamma}^{[jklmn}{}_i{}^{p]} \in \overline{\mathbf{21}}_2 \oplus \overline{\mathbf{28}}_2\,, \end{aligned} \tag{8.74}$$

The next step is to identify the different locally non-geometric R-fluxes. We identify the following spacetime fluxes

$$\begin{aligned} R^{i,jklm} &= 4\hat{\partial}^{i[j}\Omega^{klm]} - e_{\bar{i}}{}^{[j}\epsilon^{klm]inpq}\hat{\partial}_{pq}e^{\bar{i}}{}_n\,, \\ R^{ij}{}_k &= \hat{\partial}_{kl}\Omega^{ijl} - \frac{1}{72}\epsilon_{klmnpqr}\hat{\partial}^{ij}\Omega^{lmnpqr} - \frac{1}{36}\epsilon_{klmnpqr}\Omega^{lmn}\hat{\partial}^{ij}\Omega^{pqr} \\ &\quad + 4e_{\bar{i}}{}^{[i}\hat{\partial}^{j]}e^{\bar{i}}{}_k + 4\delta_k^{[i}e_{\bar{i}}{}^{j]}\hat{\partial}^l e^{\bar{i}}{}_l\,, \\ R^i &= \hat{\partial}_{jk}\Omega^{ijk} - 4\hat{\partial}^i|e| - 8e_{\bar{i}}{}^i\hat{\partial}^j e^{\bar{i}}{}_j\,, \\ R &= \epsilon_{ijklmnp}\hat{\partial}^i\Omega^{jklmnp} - 2\epsilon_{ijklmnp}\Omega^{ijk}\hat{\partial}^m\Omega^{lnp}\,, \\ R^{ijkl} &= \frac{5}{8}\hat{\partial}^{[i}\Omega^{jkl]} + \frac{1}{2}\hat{\partial}_{pq}\Omega^{pqijkl} + \frac{1}{4}\Omega^{[ijk}\hat{\partial}_{pq}\Omega^{lpq]}\,. \end{aligned} \tag{8.75}$$

The first R-flux $R^{i,jklm}$ is already known from the four-dimensional exceptional field theory, the second R-flux arises first in the six-dimensional R-flux (without the connection terms). The third R-flux is quite similar to the second R-flux, and one would be able to write down here a different choice which can be combined with the second one, which results in the following linear combination

$$\tilde{R}^{ij}{}_k = R^{ij}{}_k + \alpha R^{[i}\delta_k^{j]}\,, \tag{8.76}$$

where α can be any number. The last two fluxes are the actual new fluxes in seven dimensions. This is a result of the mirrored structure of $\mathrm{E}_{7(7)}$: The winding number of the Kaluza-Klein monopole y_i is structure-wise dual to the coordinates x^i, as well as the corresponding derivatives. The same is true for the 3-forms and the 6-forms. Therefore, the locally non-geometric R-flux R mirrors the geometric $G_7 = G_{ijklmnp}$ of SUGRA, whereas R^{ijkl} mirrors the flux G_4.

To complete the analysis of the seven-dimensional fluxes we state the mapping of the fluxes to the corresponding representations in the decomposed embedding tensor as

$$\begin{gathered} R^{i,jklm} \in \overline{\mathbf{21}}_2 \oplus \overline{\mathbf{224}}_2\,, \qquad R^{ij}{}_k \in \mathbf{7}_6 \oplus \mathbf{140}_6\,, \qquad R^i \in \mathbf{7}_6\,, \\ R^{ijkl} \in \overline{\mathbf{35}}_{10}\,, \qquad R \in \mathbf{1}_{14}\,. \end{gathered} \tag{8.77}$$

8.2 Duality Chains and Missing Momenta

In the following section, we are going to construct duality chains (as we did before in section 7.3) to find examples of backgrounds with non-geometric fluxes. To find these we have to apply U-dualities on parallelisable spaces with geometric fluxes. Note that we will refer to these spaces as “twisted tori”, even though some of those manifolds are non-compact and therefore are not typical representatives of tori. After defining those spaces and showing that they are well-defined, we will have a closer look on those twisted tori configurations as well as their membrane content. We show that some wrapping modes cannot exist due to the geometry of the space. We are going to refer to these modes as “missing wrapping modes”. This has been shown before in [90] for the example of a direct product of the Heisenberg nilmanifold with a circle. Here, we generalize this to higher dimensions and to non-compact spaces. After investigating those product spaces we are going to apply U-dualities in different directions to find backgrounds with the new non-geometric R-fluxes. The missing wrapping modes in the Heisenberg nilmanifolds become, after the U-dualities, missing momenta, which can be related to the locally non-geomertric R-fluxes.

8.2.1 Geometric flux

We start to investigate tori with geometric fluxes, also called twisted tori. Note that we have to distinguish between the trace and the trace-free part of the geometric flux.

Nilmanifold

The first space we investigate is the Heisenberg nilmanifold $\mathcal{N}_3$. It can be defined as the coset space [91]

$$\mathcal{N}_3 = \frac{G_N(\mathbb{R})}{G_N(\mathbb{Z})}\,, \tag{8.78}$$

where $G_N(\mathbb{R})$ and $G_N(\mathbb{Z})$ are defined as

$$G_N(\mathbb{F}) = \left\{ \begin{pmatrix} 1 & c & -\frac{1}{N}\,a \\ 0 & 1 & b \\ 0 & 0 & 1 \end{pmatrix} \;\middle|\; a,\, b,\, c \in \mathbb{F} \right\}\,, \tag{8.79}$$

and $\mathbb{F} \in \{\mathbb{R}, \mathbb{Z}\}$.

The name “Heisenberg nilmanifold” arises for two reasons: First, for fixed N the group $G_N(\mathbb{R})$ is isomorphic to the Heisenberg group. Second the term “nilmanifold” mirrors the property of the algebra of $G_N(\mathbb{R})$ which is a three-dimensional nilpotent algebra. Note that $G_1(\mathbb{Z})$ is isomorphic to the discrete Heisenberg group, whereas $G_N(\mathbb{R})$ is isomorphic to $\mathbb{R}^3$ for fixed N. Introducing coordinates x^1, x^2 and x^3 on the space $\mathcal{N}_3$ instead of a, b and c combined with the quotient, we find the identification

$$\left(x^1,\, x^2,\, x^3\right) \sim \left(x^1+1,\, x^2,\, x^3\right) \sim \left(x^1,\, x^2,\, x^3+1\right) \sim \left(x^1 - Nx^3,\, x^2+1,\, x^3\right)\,. \tag{8.80}$$

A possible choice of well-defined 1-forms is given by

$$\begin{aligned} e^{\bar{1}} &= dx^1 + Nx^2 dx^3 \,, \\ e^{\bar{2}} &= dx^2 \,, \\ e^{\bar{3}} &= dx^3 \,. \end{aligned} \tag{8.81}$$

Therefore, $\mathcal{N}_3$ is a parallelisable space and we can use it as a starting point for duality chains in exceptional field theories. For the 1-forms we find that

$$de^{\bar{1}} = Ne^{\bar{2}} \wedge e^{\bar{3}} \,, \qquad de^{\bar{2}} = de^{\bar{3}} = 0 \,. \tag{8.82}$$

Using the definition of the geometric flux

$$de^{\bar{i}} = T_{\bar{j}\bar{k}}{}^{\bar{i}} e^{\bar{j}} \wedge e^{\bar{k}} \,, \tag{8.83}$$

we can see by direct inspection that the description (8.81) results in the flux $T_{23}{}^{1} = N$. Due to the definition of vector fields on the manifold, $\mathcal{N}_3$ is a paralellisable space.

In the upcoming discussions in this section, our spaces will be direct sums between n-dimensional tori T^n and the Heisenberg nilmanifold $\mathcal{N}_3$. We will denote these spaces as

$$\mathcal{N}_3^n \equiv \mathcal{N}_3 \times T^n \,, \tag{8.84}$$

the dimensionality of those spaces is $\dim\left(\mathcal{N}_3^n\right) = 3 + n$. In the analysis of the spaces, we are going to label the coordinates of $\mathcal{N}_3$ with the first coordinates x^1, x^2 and x^3 as above, whereas the labels of the tori T^n will be labeled with $(x^4, \ldots, x^{n+3})$. The metric is the usual metric of the twisted torus plus the metric of the untwisted torus given by

$$ds^2 = \left(dx^1 + Nx^2 dx^3\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \ldots + \left(dx^{n+3}\right)^2 \,. \tag{8.85}$$

This agrees with the definitions above and as in section 7.3.

Non-unimodular geometric flux

The second space relevant for the geometric flux is the two-dimensional space we are denoting with $\mathcal{N}_2$. The space has the flux $T_{12}{}^{1} = N$, a possible choice of globally well-defined 1-forms is

$$\begin{aligned} e^{\bar{1}} &= dx^1 + Nx^1 dx^2 \,, \\ e^{\bar{2}} &= dx^2 \,. \end{aligned} \tag{8.86}$$

Their pullbacks are

$$de^{\bar{1}} = Ne^{\bar{1}} \wedge e^{\bar{2}} \,, \qquad de^{\bar{2}} = 0 \,, \tag{8.87}$$

which results exactly in expected geometric flux $T_{12}{}^{1} = N$.

Due to the fact that the only possible differential form of the second de Rham cohomology class is $e^{\bar{1}} \wedge e^{\bar{2}}$ which is an exact form, the second de Rham cohomology vanishes, i.e.

$$H^2_{dR}\left(\mathcal{N}_2\right) = 0\,. \tag{8.88}$$

Using the theorem that for any d-dimensional, compact, oriented and connected manifold M the d-th cohomology has to be

$$H^d\left(M\right) = \mathbb{R}\,, \tag{8.89}$$

we can assume that $\mathcal{N}_2$ is non-compact [92].[2] One example of such a space is the solvmanifold

$$\mathcal{S}_2 = \frac{S_N(\mathbb{R})}{\Lambda}\,, \tag{8.90}$$

where $S_N(\mathbb{R})$ is the matrix group

$$S_N(\mathbb{R}) = \left\{ \begin{pmatrix} e^{x^2} & e^{(1+N)x^2}\,x^1 \\ 0 & e^{(1+N)x^2} \end{pmatrix} \middle| \left(x^1,\, x^2\right) \in \mathbb{R}^2 \right\}\,. \tag{8.91}$$

The discrete subgroup Λ is defined as

$$\Lambda = \left\{ \begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix} \middle| b \in \mathbb{Z} \right\}\,, \tag{8.92}$$

so that the right-quotient of $S_N(\mathbb{R})$ by Λ leads to the identifications

$$\left(x^1, x^2\right) \sim \left(x^1 + e^{-Nx^2}, x^2\right)\,. \tag{8.93}$$

This space is non-compact due to the fact that we do not have any restrictions or identifications in the x^2 direction. The space itself is a cylinder with growing radius when x^2 grows. Note that due to the non-compactness of the space we do not have any quantization constrains on N, which motivates the name as non-unimodular geometric flux. This can be also seen from a topological perspective: While for the geometric flux the value of N counts the number of "twists" and therefore, different values of N can be distinguished topologically, this is not the case for the non-unimodular geometric flux. Any non-zero value of N is topologically equivalent and can only be distinguished by introducing a length scale for x^1, for example by a metric. Later, we are going to use direct sums between the space $\mathcal{N}_2$ and tori T^n. We are going to denote these spaces with

$$\mathcal{N}_2^m \equiv \mathcal{N}_2 \times T^m\,. \tag{8.94}$$

In here, the dimensionality of such spaces is $\dim\left(\mathcal{N}_2^n\right) = 2 + n$. Again, the coordinates for $\mathcal{N}_2$ are denoted by x^1 and x^2, whereas the coordinates of the torus T^n are denoted with $(x^3, \ldots, x^{n+2})$. The metric of the space is

$$ds^2 = \left(dx^1 + Nx^1 dx^2\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \ldots + \left(dx^{m+2}\right)^2\,. \tag{8.95}$$

---

[2] The space $\mathcal{N}_2$ is orientable and connected, therefore, the only assumption for the space which might fail is the assumption of compactness.

8.3 Wrapping States of Twisted Tori

In this section we will analyze the consequences of the geometrical fluxes for possible wrapping modes. Those wrapping modes are classified by the homology classes of the spaces. Therefore, we have to compute the second and the fifth homology group which are linked to the M2 and to the M5-brane. “Missing” cylces in the homology imply that a wrapping mode is not allowed. Missing wrapping modes will lead to missing momenta modes in the dualized theory. We will compute the different cohomology classes, and use the Poincaré duality to find the homology class.

Wrapping States of $\mathcal{N}_3^1$

The first space we examine is the four-dimensional space $\mathcal{N}_3^1 = \mathcal{N}_3 \times S^1$. Let us compute all the de Rham cohomology classes to present the main calculations, in higher dimensions we will leave out the details of the computations. We start with the possible 1-forms

$$\begin{aligned} e^{\bar{1}} &= dx^1 + Nx^2 dx^3 \,, \\ e^{\bar{2}} &= dx^2 \,, \\ e^{\bar{3}} &= dx^3 \,, \\ e^{\bar{4}} &= dx^4 \,. \end{aligned} \tag{8.96}$$

As we can see, all of them are closed, but not all of them are exact:

$$de^{\bar{1}} = N dx^2 \wedge x^3 \neq 0 \,, \quad de^{\bar{2}} = de^{\bar{3}} = de^{\bar{4}} = 0 \,. \tag{8.97}$$

Therefore, the first de Rham cohomology is

$$H^1_{dR}\left(\mathcal{N}_2\right) = \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R} \,. \tag{8.98}$$

For the second de Rham cohomology (which corresponds to the possible wrapping modes) we have to analyze the different two-forms α, which are closed, but not exact.[3] The two-form $e^{\bar{2}} \wedge e^{\bar{3}}$ is not exact because it holds that

$$\frac{1}{N} de^{\bar{1}} = e^{\bar{2}} \wedge e^{\bar{3}} \,. \tag{8.99}$$

On the other hand the two-form $e^{\bar{1}} \wedge e^{\bar{4}}$ is not closed

$$d\left(e^{\bar{1}} \wedge e^{\bar{4}}\right) = N e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \,. \tag{8.100}$$

Therefore, the de Rham cohomology is

$$H^2_{dR}(\mathcal{N}_3^1) = \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R} \,. \tag{8.101}$$

---

[3] A form is closed if $d\alpha = 0$ holds, whereas a form α is exact if there exist a form β such that $\alpha = d\beta$.

Since we need to compute the homology, we could apply the Poincaré duality and get

$$H_2(\mathcal{N}_3^1) = \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R} \oplus \mathbb{R} \,. \tag{8.102}$$

This is unfortunately a rather naive approach, since the wrapping modes are determined by the integer homology groups. For the integer homology group we also have to take into account the torsion subgroups generated by the (x^2x^3) and (x^1x^4) cycles. It was shown in [91] that this leads to the integer homology groups

$$H_2(\mathcal{N}_3^1, \mathbb{Z}) = \mathbb{Z} \oplus \mathbb{Z} \oplus \mathbb{Z} \oplus \mathbb{Z} \oplus \mathbb{Z}_{k_1} \oplus \mathbb{Z}_{k_2} \,, \tag{8.103}$$

with k_1 and k_2 are integers determining the rank of the torsion subgroup. However, $H_2(\mathcal{N}_3^1, \mathbb{Z})$ cannot have torsion because $\mathcal{N}_3^1$ is closed and oriented. Therefore, no M2-branes are allowed wrapping the two-cycles (x^1x^4) and (x^2x^3), which leads to $w^{14} = w^{23} = 0$.

Another possibility to understand the missing wrapping modes is to use the Freed-Witten anomaly condition. The Freed-Witten anomaly states that in Type IIB string theory we cannot have D3-branes on a three-dimensional torus with H-flux. Applying a T-duality transformation in the x^1 direction, and assuming that x^4 is a vanishing cycle, the missing D3-brane in Type IIB is dual to a missing M2-brane in M-theory, wrapping the x^2x^3-directions.

Wrapping States of $\mathcal{N}_3^1$

The five-dimensional space $\mathcal{N}_3^2 = \mathcal{N}_3 \times T^2$ is analogous to the space $\mathcal{N}_3^1$. The set of well-defined one-forms is given as

$$\begin{aligned} e^{\bar{1}} &= dx^1 + N x^2 dx^3 \,, \\ e^{\bar{2}} &= dx^2 \,, \\ e^{\bar{3}} &= dx^3 \,, \\ e^{\bar{4}} &= dx^4 \,, \\ e^{\bar{5}} &= dx^5 \,. \end{aligned} \tag{8.104}$$

Here, we can see that the forms $e^{\bar{2}} \wedge e^{\bar{3}}$, $e^{\bar{1}} \wedge e^{\bar{4}}$ and $e^{\bar{1}} \wedge e^{\bar{5}}$ are not part of the de Rham cohomology since they satisfy

$$\frac{1}{N} de^{\bar{1}} = e^{\bar{2}} \wedge e^{\bar{3}} \,, \qquad d\left(e^{\bar{1}} \wedge e^{\bar{4}}\right) = N e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \,, \qquad d\left(e^{\bar{1}} \wedge e^{\bar{5}}\right) = N e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{5}} \,. \tag{8.105}$$

Therefore, the integer homology group becomes

$$H_2(\mathcal{N}_3^2, \mathbb{Z}) = \mathbb{Z}^7 \oplus \mathbb{Z}_{k_1} \oplus \mathbb{Z}_{k_2} \oplus \mathbb{Z}_{k_3} \,, \tag{8.106}$$

and therefore, the wrapping modes $w^{23} = w^{14} = w^{15} = 0$ vanish. In five compact dimensions we also have to look at the fifth cohomology group because we are interested in wrapping modes of the M5-brane. The only existing component is the five-form $e^{\bar{1}} \wedge e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \wedge e^{\bar{5}}$. This form is closed because

$$d\left(e^{\bar{1}} \wedge e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \wedge e^{\bar{5}}\right) = N e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \wedge e^{\bar{5}} = 0 \,. \tag{8.107}$$

Here, we cannot construct any four-form which might lead to this five-form. Therefore, the fifth de Rham cohomology is $H^5_{dR}(\mathcal{N}_3^2) = \mathbb{R}$, and therefore, the fifth integer homology group becomes $H_5(\mathcal{N}_3^2, \mathbb{Z}) = \mathbb{Z}$ which leads to $w^{12345} \neq 0$.

Wrapping States of $\mathcal{N}_3^3$

In six dimensions we are mainly interested in the space $\mathcal{N}_3^3 = \mathcal{N}_3 \times T^3$. Therefore, we are interested on the consequences of the geometry on the allowed wrapping modes of the M2-brane and the M5-brane. From the four-dimensional space $\mathcal{N}_3^1$ we already know that it is sufficient to compute the de Rham cohomolgy - the integer homology group is the same up to torsion group elements. To find the de Rham cohomology group let us use the well-defined one-forms from (8.81). Additionally, we use

$$e^{\bar{4}} = dx^4\,, \qquad e^{\bar{5}} = dx^5\,, \qquad e^{\bar{6}} = dx^6\,. \tag{8.108}$$

Most important for the cohomology group is the relation

$$de^{\bar{1}} = Ne^{\bar{2}} \wedge e^{\bar{3}}\,. \tag{8.109}$$

Using this relation, we can directly show that the two-forms $e^{\bar{1}} \wedge e^{\bar{4}}$, $e^{\bar{1}} \wedge e^{\bar{5}}$ and $e^{\bar{1}} \wedge e^{\bar{6}}$ are not closed

$$\begin{aligned} d\left(e^{\bar{1}} \wedge e^{\bar{4}}\right) &= Ne^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \neq 0\,, \\ d\left(e^{\bar{1}} \wedge e^{\bar{5}}\right) &= Ne^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{5}} \neq 0\,, \\ d\left(e^{\bar{1}} \wedge e^{\bar{6}}\right) &= Ne^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{6}} \neq 0\,. \end{aligned} \tag{8.110}$$

As a consequence of the equations above, the two-cycles (x^2x^3), (x^1x^4), (x^1x^5) and (x^1x^6) are not part of the homology $H_2\left(\mathcal{N}_3^3\right)$ or are generated from a torsion subgroup. In here, there are no wrapping modes of the M2-brane $w^{23} = w^{14} = w^{15} = w^{16} = 0$. Additionally, we are interested in the fifth homology group. From the relation (8.110), we can deduce that

$$d\left(e^{\bar{1}} \wedge e^{\bar{4}} \wedge e^{\bar{5}} \wedge e^{\bar{6}}\right) = Ne^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \wedge e^{\bar{5}} \wedge e^{\bar{6}} \neq 0\,. \tag{8.111}$$

As a result, the five-form $e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \wedge e^{\bar{5}} \wedge e^{\bar{6}}$ is exact and therefore not part of the homology (at least not of the torsion free part). All other elements are part of the homology, the only vanishing wrapping state of the M5-brane is the state $w^{23456} = 0$.

Let us note that the missing momenta modes $w^{23} = w^{23456} = 0$ can be understood from the Freed-Witten duality, performing T-dualities in x^1 or x^1, x^4 and x^5 respectively, while using the direction x^6 as M-theory direction.

Wrapping States of $\mathcal{N}_3^4$

We now consider the seven-dimensional manifold $\mathcal{N}_3^4 = \mathcal{N}_3 \times T^4$. Using the notation

$$\begin{aligned} e^{\bar{1}} &= dx^1 + Nx^2dx^3\,, \qquad e^{\bar{2}} = dx^2\,, \qquad e^{\bar{3}} = dx^3\,, \\ e^{\bar{4}} &= dx^4\,, \qquad e^{\bar{5}} = dx^5\,, \qquad e^{\bar{6}} = dx^6\,, \qquad e^{\bar{7}} = dx^7\,, \end{aligned} \tag{8.112}$$

we find the relations

$$\begin{aligned} e^{\bar{2}} \wedge e^{\bar{3}} &= \frac{1}{N} de^{\bar{1}} \,, \\ d\left(e^{\bar{1}} \wedge e^{\bar{\mu}}\right) &\neq 0 \,, \\ e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{\mu}} \wedge e^{\bar{\nu}} \wedge e^{\bar{\rho}} &= \frac{1}{N} d\left(e^{\bar{1}} \wedge e^{\bar{\mu}} \wedge e^{\bar{\nu}} \wedge e^{\bar{\rho}}\right) \,, \\ d\left(e^{\bar{1}} \wedge e^{\bar{4}} \wedge e^{\bar{5}} \wedge e^{\bar{6}} \wedge e^{\bar{7}}\right) &= N e^{\bar{2}} \wedge e^{\bar{3}} \wedge e^{\bar{4}} \wedge e^{\bar{5}} \wedge e^{\bar{6}} \wedge e^{\bar{7}} \,, \end{aligned} \tag{8.113}$$

between the different forms, where $\mu, \nu, \rho = 4, \ldots, 7$. The relations above show that all the two-forms and the five-forms above are not part of the de Rham cohomology because they are either not closed or are exact. The associated cycles must vanish in the integer homology of $\mathcal{N}_3^4$ or generate torsion subgroups. We deduce that the two-cycles $(x^2 x^3)$ and all the five-cycles listed cannot generate torsion and therefore must vanish in the homology. This means that the M2- and M5-brane wrapping modes

$$w^{23} = w^{23456} = w^{23457} = w^{23567} = w^{23467} = w^{14567} = 0 \tag{8.114}$$

must vanish. Again, we can understand such missing modes from the Freed-Witten anomaly in the IIB background obtained by dualizing along x^1 and x^7. Due to the fact the $\mathcal{N}_3^4$ is seven-dimensional, we also have to take a look at the wrapping modes from Kaluza-Klein monopoles. For our purposes it is enough to show that the wrapping modes $w_{KK}^2 = w_{KK}^3 = 0$ vanish.

As we showed above the wrapping state $w^{23567} \simeq w_{14} = 0$ vanishes. When we perform a U-duality transformation in the x^1, x^3 and x^4 direction the states w^{23567} and w_{KK}^3 are exchanged (using the relations A.38) as

$$w^{23567} \xrightarrow{U_{134}} w_{KK}^3 \,, \tag{8.115}$$

but under the map U_{134} the background stays the same. Therefore, the fact that $w^{23567} = 0$ vanishes implies that $w_{KK}^3 = 0$ must vanish as well, otherwise the background would change. Using the symmetry of the direction x^2 and x^3 of the space $\mathcal{N}_3^4$ and changing the definitions of the 1-forms accordingly, we can perform the same trick to show that the wrapping mode $w_{KK}^2 = 0$ must vanish as well.

Wrapping States of $\mathcal{N}_2^3$

We are interested in the five-dimensional space $\mathcal{N}_2^3 = \mathcal{N}_2 \times T^3$. We will construct the totally anti-symmetric locally non-geometric R-flux R^{ijklm} from this background. For this, let us examine the missing wrapping states in this space. We can directly argue that there are no possible wrapping modes in the x^2 direction allowed due to the fact that the direction is non-compact. Therefore, the wrapping modes for the M2-branes and the M5-branes $w^{12} = w^{23} = w^{24} = w^{25} = w^{12345} = 0$ must vanish.

Wrapping States of $\mathcal{N}_2^5$

The last space we examine is the space $\mathcal{N}_2^5 = \mathcal{N}_2 \times T^5$. Once again, we are using the fact that the space in the x^2 direction is not compact and therefore, we cannot have any wrapping modes in this direction. In particular, the wrapping modes $w^{12} = w^{12\mu\nu\rho} = w^2_{KK} = 0$ must vanish with $\mu, \nu, \rho = 3, \dots, 7$.

8.4 Duality Chains

8.4.1 Review of the four-dimensional duality chain

We reviewed the duality chain in four dimensions already in section 7.3.2. Let us focus on the effect of U-duality transformations on the missing wrapping modes of the membranes. The duality chain we are using is

$$T_{23}{}^1 \xrightarrow{U_{234}} R^{4,1234}\,, \tag{8.116}$$

where U_{234} denotes the U-duality transformation along x^2, x^3 and x^4.

In section 8.3, we showed that in background with $T_{23}{}^1$ the wrapping modes satisfy $w^{23} = w^{14} = 0$. The proposed U-duality transformation U_{234} exchanges wrapping modes and momenta modes in the following way:

$$w^{23} \to p_4\,, \qquad w^{24} \to p_2\,, \qquad w^{34} \to p_2\,. \tag{8.117}$$

The missing wrapping mode $w^{23} = 0$ in the background with geometric flux $T_{23}{}^1 = N$ becomes a missing momenta mode $p_4 = 0$ in the background with $R^{4,1234} = N$. It was conjectured in [90] that this can be written as

$$R^{i,jklm} p_i = 0\,. \tag{8.118}$$

In the following sections we describe the different duality chains necessary to find the different R-fluxes in higher-dimensional compactified spaces. We will see that similar conjectures can be found for all mixed-symmetry R-fluxes.

8.4.2 Five-dimensional duality chain

In the five-dimensional space, we were able to find two different types of R-fluxes: the R-flux $R^{i,jklm}$ which is the same R-flux as in four dimensions, and the totally anti-symmetric R-flux:

$$R^{ijklm} = 20\hat{\partial}^{[ij}\Omega^{klm]} - 2\epsilon^{ijklm}\hat{\partial}^z |e|\,. \tag{8.119}$$

Duality chain for $R^{i,jklm}$

In five dimensions, we are able to construct the R-flux $R^{4,1234} = N$ in the same way as in four dimensions. For completeness, will show that the same conjecture (8.118) can be applied in higher dimensions for an R-flux $R^{i,123i} = N$ as well (with $i = 4, \dots d$).

Starting from the background with geometric flux $T_{23}{}^{1} = N$ given by

$$\begin{aligned} ds^2 =& \left(dx^1 + Nx^2 dx^3\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 , \\ C_{ijk} =& \Omega^{ijk} = 0 \qquad \text{with } i,j,k = 1, \dots, 5 , \end{aligned} \tag{8.120}$$

we apply a U-duality transformation in the directions x^2, x^3 and x^i with $i = 4, \dots d$ (in this case $d = 5$, but this is true in higher dimensions as well). Using the U-duality transformation rules A.2, we find that

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^4\right)^2 + (dx)^5 , \\ \Omega^{12i} &= -N\tilde{x}_{3i} , \end{aligned} \tag{8.121}$$

where we used

$$x^2 \xrightarrow{U_{23i}} \tilde{x}_{3i} . \tag{8.122}$$

This background has a locally non-geometric R-flux $R^{i,123i} = N$. As last step, we check that the missing wrapping modes transform in the right way and become missing momenta modes. As we saw in the previous section the cycle (x^2x^3) vanished independently of the dimension d of the space $\mathcal{N}_3^{d-3}$. Therefore, under the U-duality transformation U_{23i} we always find that the corresponding wrapping mode w^{23} becomes

$$w^{23} \xrightarrow{U_{23i}} p_i , \tag{8.123}$$

and as a a result, $p_i = 0$. Therefore, the conjecture (8.118) is in arbitrary dimensions true as long as the space $\mathcal{N}_3^{d-3}$ is well defined.

Duality chain for R^{ijklm}

For the totally anti-symmetric R-flux R^{ijklm} we have to use a different duality chain: The flux lies in the $\overline{\mathbf{16}}$-representation of the embedding tensor while the flux $R^{i,jklm}$ is part of the $\overline{\mathbf{144}}$. Therefore, we use the non-unimodular space $\mathcal{N}_2^3$ as a starting point which is in the $\overline{\mathbf{16}}$ as well. This space is described by

$$\begin{aligned} ds^2 =& \left(dx^1 + Nx^1 dx^2\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + + \left(dx^5\right)^2 , \\ C_{ijk} =& \Omega^{ijk} = 0 \qquad \text{with } i,j,k = 1, \dots, 5 , \end{aligned} \tag{8.124}$$

which contains a geometric flux

$$T_{12}{}^{1} = N . \tag{8.125}$$

We begin with a U-duality in the x^1, x^2 and the x^3 direction. Using the formulae (A.37) we find that the non-geometric three-form gets one non-zero component

$$e^{\bar{1}}{}_2 = Nx^1 \longrightarrow \Omega^{134} = Nx^1 \,, \tag{8.126}$$

and therefore we find for the background

$$\begin{aligned} ds^2 &= dx^1 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + + \left(dx^5\right)^2 \,, \\ \Omega^{134} &= Nx^1 \,. \end{aligned} \tag{8.127}$$

As a result, we have the background with a globally non-geometric Q-flux ${Q_1}^{134} = N$. To find the locally non-geometric R-flux, we have to dualize the coordinate x^1. This can be done via a U-duality transformation along the x^1, x^4 and x^5 directions. We can check that this changes $x^1 \to \tilde{x}_{45}$, while leaving the metric and the non-geometric 3-form invariant:

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 \,, \\ \Omega^{134} &= N\tilde{x}_{25} \,. \end{aligned} \tag{8.128}$$

As a result, we have a background with the a non-vanishing R-flux with components

$$\begin{aligned} R^{2,1345} &= -N \,, \\ R^{5,1234} &= N \,. \end{aligned} \tag{8.129}$$

Due to the minus sign the totally anti-symmetric component does not vanish and we find

$$R^{12345} = 2N \,. \tag{8.130}$$

Therefore, we find the duality chain which results in a background with non-vanishing R-flux R^{ijklm}:

$${T_{12}}^1 \xrightarrow{U_{234}} {Q_1}^{134} \xrightarrow{U_{125}} R^{2,3145} = -R^{5,1234} \,. \tag{8.131}$$

To finish the analysis of the duality chain, we examine the transformation behavior of the missing wrapping states. As we showed in 8.3, the wrapping modes $w^{12} = w^{12345} = 0$ vanish in particular. Using the U-duality transformation law in appendix A.2, we find for momenta in the duality chain

$$\begin{aligned} w^{12} &\xrightarrow{U_{234}} w^{12} \xrightarrow{U_{125}} p_5 \,, \\ w^{12345} &\xrightarrow{U_{234}} w^{15} \xrightarrow{U_{125}} p_2 \,. \end{aligned} \tag{8.132}$$

Thus, we find two missing momenta modes $p_2 = p_5 = 0$, which shows that also in this case the conjecture

$$R^{i,jklm} p_i = 0 \,, \tag{8.133}$$

is fulfilled. This duality chain also gives a hint for the appearance of the flux in five dimensions. To find such a background, we have to perform two U-dualities which only share one direction. Therefore, we need to have at least five dimensions to construct this kind of R-flux.

8.4.3 Six-dimensional duality chain

As we saw in section 8.1.3 we get a new non-geometric R-flux in six dimensions with mixed symmetries given by

$$R^{ij,klmnpq} = 6\hat{\partial}^{[klmnp}\Omega^{q]ij} + 10\hat{\partial}^{ij}\Omega^{klmnpq} + 20\Omega^{[klm}\hat{\partial}^{|ij|}\Omega^{npq]} \,. \tag{8.134}$$

We are now going to construct a background which has this new R-flux. As we saw in the last section, it is crucial to start in the right part of the embedding tensor. Because the flux $R^{ij,klmnpq}$ is part of the $\overline{\mathbf{351}}$ we start again with a background which has the geometric flux $T_{23}{}^{1}$. Therefore, we start with the six-dimensional twisted torus $\mathcal{N}_3^3$ which is described by

$$\begin{aligned} ds^2 &= \left(dx^1 + Nx^2 dx^3\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + + \left(dx^5\right)^2 + \left(dx^6\right)^2 \,, \\ C_{ijk} &= \Omega^{ijk} = C_{ijklmn} = \Omega^{ijklmn} = 0 \qquad \text{for} \qquad i,j,\ldots = 1,\ldots,6 \,. \end{aligned} \tag{8.135}$$

To find a background with the right non-geometric R-flux, we perform two U-duality transformations, each in three dimensions. Note that duality chains are not unique and we are not providing here a framework how to find such duality chains. We have to choose a suitable ansatz to find the correct transformation behavior. Let us start with performing a U-duality transformation along the directions x^2, x^3 and x^4. This is the same duality transformation we used in the four-dimensional case and therefore, we find the background

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 + \left(dx^6\right)^2 \,, \\ \Omega^{124} &= -N\tilde{x}_{34} \,, \end{aligned} \tag{8.136}$$

with the non-geometric R-flux $R^{4,1234} = N$. Looking at the structure of the flux $R^{ij,klmnpq}$, a straightforward way to find the right background is to perform a U-duality in the directions x^1, x^5 and x^6 which takes

$$\tilde{x}_{34} \longrightarrow \tilde{x}_{13456} \,, \tag{8.137}$$

while leaving the metric and the three-form invariant. Therefore, we find for the background

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 + \left(dx^6\right)^2 \,, \\ \Omega^{124} &= -N\tilde{x}_{13456} \,, \end{aligned} \tag{8.138}$$

which has now become a background carrying the new R-flux

$$R^{14,123456} = 6\partial^{[12345}\Omega^{6]14} = N \,. \tag{8.139}$$

We find thus a duality chain which results in the new six-dimensional R-flux

$$T_{23}{}^{1} \xrightarrow{U_{234}} R^{4,1234} \xrightarrow{U_{156}} R^{14,123456} \,. \tag{8.140}$$

To complete the analysis of the transformation behaviour of the space $\mathcal{N}_3^3$, we recall that in particular the wrapping modes of the M2-brane and of the M5-brane vanish, i.e. $w^{23} = w^{23456} = 0$. These wrapping modes are exchanged under the U-duality transformation with corresponding momenta

$$\begin{aligned} &w^{23} \xrightarrow{U_{234}} p_4 \xrightarrow{U_{156}} p_4 \,, \\ &w^{23456} \xrightarrow{U_{234}} w^{56} \xrightarrow{U_{156}} p_1 \,, \end{aligned} \tag{8.141}$$

and therefore, we have two missing momenta, namely $p_4 = 0$ and $p_1 = 0$. We find then a natural extension of the conjecture (8.118), which is

$$R^{ij,klmnpq} p_i = 0 \,. \tag{8.142}$$

Note that after the first U-duality transformation, we again find the already known R-flux conjecture

$$R^{i,jklm} p_i = 0 \,. \tag{8.143}$$

8.4.4 Seven-dimensional duality chain

Finally, we conclude with the seven-dimensional duality chains. The exceptional field theory $\mathrm{E}_{7(7)}$ has several new R-fluxes

$$\begin{aligned} R^{ij}{}_k &= \hat\partial_{kl}\Omega^{ijl} - \frac{1}{72}\epsilon_{klmnpqr}\hat\partial^{ij}\Omega^{lmnpqr} - \frac{1}{36}\epsilon_{klmnpqr}\Omega^{lmn}\hat\partial^{ij}\Omega^{pqr} \\ &\quad + 4e_{\bar i}{}^{[i}\hat\partial^{j]}e^{\bar i}{}_k + 4\delta_k^{[i}e_{\bar i}{}^{j]}\hat\partial^l e^{\bar i}{}_l \,, \\ R^i &= \hat\partial_{jk}\Omega^{ijk} - 4e_{\bar i}{}^j\hat\partial^i e^{\bar i}{}_j - 8e_{\bar i}{}^i\hat\partial^j e^{\bar i}{}_j \,, \\ R &= \epsilon_{ijklmnp}\hat\partial^i\Omega^{jklmnp} - 2\epsilon_{ijklmnp}\Omega^{ijk}\hat\partial^m\Omega^{lnp} \,, \\ R^{ijkl} &= \hat\partial^{[i}\Omega^{jkl]} + \frac{4}{5}\hat\partial_{pq}\Omega^{pqijkl} + \frac{2}{5}\Omega^{[ijk}\hat\partial_{pq}\Omega^{l]pq} \,. \end{aligned} \tag{8.144}$$

Note that the first flux can be redefined as a linear combination of the first and the second flux as

$$\tilde R^{ij}{}_k = R^{ij}{}_k + \alpha R^{[i}\delta_k^{j]} \,. \tag{8.145}$$

Using the conjectures and assuming they must also hold in seven dimensions, we will find that $\alpha = -2$. We present two different duality chains which lead to two new conjectures on missing momenta.

Duality chain for $R^{ij}{}_j$ and R^i

First, we are going to present a duality chain which results in a flux with $R^{ij}{}_j \neq 0$ or $R^i \neq 0$. Again, we start with the space $\mathcal{N}_2^5$ which has a non-unimodular flux $T_{12}{}^1 = N$.

The background is given by

$$\begin{aligned} ds^2 =& \left(dx^1 + Nx^1 dx^2\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 + \left(dx^6\right)^2 + \left(dx^7\right)^2 \,, \\ C_{ijk} =& \Omega^{ijk} = C_{ijklmn} = \Omega^{ijklmn} = 0 \,, \end{aligned} \tag{8.146}$$

with $i, j, k, l, m, n = 1, \ldots, 7$. As before, we start by performing two U-duality transformations. The first one along the directions x^2, x^3 and x^4, and the second one along the directions x^1, x^2 and x^5. This leads to the non-geometric background

$$\begin{aligned} ds^2 =& \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + + \left(dx^5\right)^2 + \left(dx^6\right)^2 + \left(dx^7\right)^2 \,, \\ \Omega^{134} =& N\tilde{x}_{25} \,. \end{aligned} \tag{8.147}$$

As we already showed in section 8.4.2, this space has the non-geometric R-flux $R^{5,1234} = -R^{2,1345} = N$ and is missing the corresponding momentum modes

$$R^{i,jklm} p_i = 0 \,. \tag{8.148}$$

As a next step, we perform a U-duality transformation along the directions x^2, x^6 and x^7. Using again the transformation laws for U-duality transformations, we find the background

$$\begin{aligned} ds^2 =& \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^5\right)^2 + \left(dx^6\right)^2 + \left(dx^7\right)^2 \,, \\ \Omega^{123467} =& N\tilde{x}_{25} \,. \end{aligned} \tag{8.149}$$

Using the definitions of the R-flux (8.144), we find that this background has the flux

$$R_5^{25} = N \,. \tag{8.150}$$

Analyzing the transformation of the missing wrapping modes $w^{12} = w^2_{KK} = 0$ of the space $\mathcal{N}_2^5$, we see that this leads to missing momenta modes $p_2 = p_5 = 0$ which can be seen in

$$\begin{aligned} & w^{12} \xrightarrow{U_{234}} w^{12} \xrightarrow{U_{125}} p_5 \xrightarrow{U_{267}} p_5 \,, \\ & w^2_{KK} \xrightarrow{U_{234}} w_{34} \simeq w^{12567} \xrightarrow{U_{125}} w_{67} \xrightarrow{U_{267}} p_2 \,. \end{aligned} \tag{8.151}$$

We find thus for the non-geometric R-flux $R^{ij}{}_k$ the conjecture

$$R^{ij}{}_k p_i = \tilde{R}^{ij}{}_k p_i = 0 \,. \tag{8.152}$$

As a next step, let us perform a U-duality transformation along the directions x_1, x_3 and x_4. This duality transformation maps the coordinate $\tilde{x}_{25} \to \tilde{x}_{12345} \simeq \tilde{x}^{67}$, and therefore, we get the background

$$\begin{aligned} ds^2 =& \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \ldots + \left(dx^7\right)^2 \,, \\ \Omega^{123467} =& N\tilde{x}^{67} \,. \end{aligned} \tag{8.153}$$

This background has two non-vanishing R-fluxes, namely R^i and $R^{ij}{}_k$:

$$R^2 = N\,, \qquad R^{26}{}_6 = R^{27}{}_7 = N\,. \tag{8.154}$$

Using the linear combination of the R-fluxes (8.145) we find

$$\begin{aligned} &\tilde{R}^{2i}{}_i = -N \qquad \text{for} \quad i = 1, \ldots, 5\,, \\ &\tilde{R}^{26}{}_6 = \tilde{R}^{27}{}_7 = 0\,. \end{aligned} \tag{8.155}$$

Our duality chain is now given by

$$T_{12}{}^1 \xrightarrow{U_{234}} Q_2{}^{234} \xrightarrow{U_{125}} R^{5,1234} = -R^{2,1345} \xrightarrow{U_{267}} R^{25}{}_5 \xrightarrow{U_{134}} R^2 = R^{26}{}_6 = R^{27}{}_7\,. \tag{8.156}$$

As in the cases before, we have to analyze the transformation behavior of the missing wrapping modes $w^2_{KK} = w^{12} = w^{12567} = w^{12467} = w^{12367} = 0$ under the duality transformations. These are given by

$$\begin{aligned} &w^{12567} \xrightarrow{U_{234}} w^{12567} \xrightarrow{U_{125}} w^{12567} \xrightarrow{U_{267}} w^{34} \xrightarrow{U_{134}} p_1\,, \\ &w^2_{KK} \xrightarrow{U_{234}} w_{34} \simeq w^{12567} \xrightarrow{U_{125}} w_{67} \xrightarrow{U_{267}} p_2 \xrightarrow{U_{134}} p_2\,, \\ &w^{12467} \xrightarrow{U_{234}} w^{12467} \xrightarrow{U_{125}} w^{12467} \xrightarrow{U_{267}} w^{14} \xrightarrow{U_{134}} p_3\,, \\ &w^{12367} \xrightarrow{U_{234}} w^{12367} \xrightarrow{U_{125}} w^{12367} \xrightarrow{U_{267}} w^{13} \xrightarrow{U_{134}} p_4\,, \\ &w^{12} \xrightarrow{U_{234}} w^{12} \xrightarrow{U_{125}} p_5 \xrightarrow{U_{267}} p_5 \xrightarrow{U_{134}} p_5\,. \end{aligned} \tag{8.157}$$

For this non-geometric background, we find that our conjecture still holds, namely

$$\tilde{R}^{ij}{}_k p_i = 0\,. \tag{8.158}$$

Duality chain for R^{ijkl} and R

To construct the fluxes R^{ijkl} and R, we start with the space $\mathcal{N}_3^4$. This background is given by

$$\begin{aligned} ds^2 &= \left(dx^1 + N x^2 dx^3\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 + \left(dx^6\right)^2 + \left(dx^7\right)^2\,, \\ C_{ijk} &= \Omega^{ijk} = C_{ijklmn} = \Omega^{ijklmn} = 0\,, \end{aligned} \tag{8.159}$$

with $i, j, k, l, m, n = 1, \ldots, 7$. The geometric flux is given as

$$T_{23}{}^1 = N\,. \tag{8.160}$$

We perform a U-duality transformation along the six directions x^2, x^3, x^4, x^5, x^6, x^7. We find the background

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 + \left(dx^6\right)^2 + \left(dx^7\right)^2\,, \\ \Omega^{124567} &= N \tilde{x}_{34567}\,. \end{aligned} \tag{8.161}$$

This background yields a new non-geometric R-flux, namely

$$R^{4567} = N\,. \tag{8.162}$$

To find a similar conjecture as in the previous sections, we have to track the transformation behavior from the missing wrapping modes. We find that

$$w^{23567} \longrightarrow p_4\,, \qquad w^{23467} \longrightarrow p_5\,, \qquad w^{23457} \longrightarrow p_6\,, \qquad w^{23456} \longrightarrow p_7\,, \tag{8.163}$$

so after the U-duality transformation U_{234567} we find four missing momenta $p_4 = p_5 = p_6 = p_7 = 0$. This corresponds exactly to

$$R^{ijkl} p_l = 0\,. \tag{8.164}$$

To find the last R-flux, we perform a U-duality transformation along the directions x^1, x^2 and x^3. Under this duality only the coordinates $\tilde{x}_{34567} \longrightarrow \tilde{x}_3$ are exchanged, leaving the metric and the six-form invariant, and therefore, we find the background

$$\begin{aligned} ds^2 &= \left(dx^1\right)^2 + \left(dx^2\right)^2 + \left(dx^3\right)^2 + \left(dx^4\right)^2 + \left(dx^5\right)^2 + \left(dx^6\right)^2 + \left(dx^7\right)^2\,, \\ \Omega^{124567} &= -N\tilde{x}_3\,. \end{aligned} \tag{8.165}$$

This space has the flux

$$R = N\,. \tag{8.166}$$

Summarizing the duality chain, we find

$$T_{23}{}^{1} \xrightarrow{U_{234567}} R^{4567} \xrightarrow{U_{123}} R^{1234567} = R\,. \tag{8.167}$$

After the U-duality transformation U_{123}, the momenta $p_4 = p_5 = p_6 = p_7 = 0$ still vanish. Looking at the vanishing wrapping modes of the M2-brane $w^{23} = 0$ and the vanishing of the KK-monopole $w^2_{KK} = w^3_{KK} = 0$, they also become missing momenta

$$\begin{aligned} w^{23} &\xrightarrow{U_{234567}} w^{23} \xrightarrow{U_{123}} p_1\,, \\ w^2_{KK} &\xrightarrow{U_{234567}} w^{13} \xrightarrow{U_{123}} p_2\,, \\ w^3_{KK} &\xrightarrow{U_{234567}} w^{12} \xrightarrow{U_{123}} p_3\,. \end{aligned} \tag{8.168}$$

Therefore, all momenta modes are now missing $p_1 = p_2 = p_3 = p_4 = p_5 = p_6 = p_7 = 0$. This can be summarized with the conjecture

$$R p_i = 0\,. \tag{8.169}$$

8.5 Summary of R-Flux Backgrounds

Let us conclude with a short summary of the possible R-fluxes in seven dimensions and their consequences. We have seen that in seven dimensions, we have a large variety of different R-fluxes

$$\begin{aligned} R^{i,jklm} &= 4\hat{\partial}^{i[j}\Omega^{klm]} - e_{\bar{i}}{}^{[j}\epsilon^{klm]inpq}\hat{\partial}_{pq}e^{\bar{i}}{}_n\,, \\ R^{ij}{}_k &= \hat{\partial}_{kl}\Omega^{ijl} - \frac{1}{72}\epsilon_{klmnpqr}\hat{\partial}^{ij}\Omega^{lmnpqr} - \frac{1}{36}\epsilon_{klmnpqr}\Omega^{lmn}\hat{\partial}^{ij}\Omega^{pqr} \\ &\quad + 4e_{\bar{i}}{}^{[i}\hat{\partial}^{j]}e^{\bar{i}}{}_k + 4\delta_k^{[i}e_{\bar{i}}{}^{j]}\hat{\partial}^l e^{\bar{i}}{}_l\,, \\ R^i &= \hat{\partial}_{jk}\Omega^{ijk} - 4e_{\bar{i}}{}^j\hat{\partial}^i e^{\bar{i}}{}_j - 8e_{\bar{i}}{}^i\hat{\partial}^j e^{\bar{i}}{}_j\,, \\ R &= \epsilon_{ijklmnp}\hat{\partial}^i\Omega^{jklmnp} - 2\epsilon_{ijklmnp}\Omega^{ijk}\hat{\partial}^m\Omega^{lnp}\,, \\ R^{ijkl} &= \frac{5}{8}\hat{\partial}^{[i}\Omega^{jkl]} + \frac{1}{2}\hat{\partial}_{pq}\Omega^{pqijkl} + \frac{1}{4}\Omega^{[ijk}\hat{\partial}_{pq}\Omega^{l]pq}\,. \end{aligned} \tag{8.170}$$

The first R-flux $R^{i,jklm}$ reduces to the R-flux R^{ijk} in the NS-NS sector of string theory, if one reduces the SUGRA notation to Type IIA string theory. Each of the other R-fluxes involves dual coordinates and therefore, has to involve wrapping modes of D-branes, or must include non-geometric field associated to the R-R sector of string theory. As far as we know, their string theory notation is not yet known.

We also presented a way how to construct background which contain these R-fluxes using U-duality transformations on product spaces of nil/solvmanifolds with tori. These can be summarized as

$$\begin{aligned} &T_{12}{}^1 \xrightarrow{U_{234}} Q_2{}^{234} \xrightarrow{U_{125}} R^{5,1234} = -R^{2,1345} \xrightarrow{U_{267}} R^{25}{}_5 \xrightarrow{U_{134}} R^2 = R^{26}{}_6 = R^{27}{}_7\,, \\ &T_{23}{}^1 \xrightarrow{U_{234}} R^{4,1234} \xrightarrow{U_{156}} R^{14}{}_7\,, \\ &T_{23}{}^1 \xrightarrow{U_{234567}} R^{4567} \xrightarrow{U_{123}} R^{1234567}\,. \end{aligned} \tag{8.171}$$

In those product spaces, we can identify wrapping modes which must vanish due to the topology of the spaces. After applying duality transformations in such spaces, those missing wrapping modes of branes might become missing momenta depending on the type of R-flux. Those missing momenta are governed by the conjectures,

$$\begin{aligned} R^{i,jklm}\,p_i &= 0\,, \\ \left(R^{ij}{}_k - 2R^{[i}\delta_k^{j]}\right)p_i &= 0\,, \\ R^{ijkl}\,p_i &= 0\,, \\ R\,p_i &= 0\,. \end{aligned} \tag{8.172}$$

Chapter 9

Non-Associativity in an R-Flux Background

In this chapter we focus on an interesting consequence of compactifications on higher-dimensional tori and having T-duality as a symmetry: Having a string compactified (open or closed) on a three-dimensional torus in combination with an H-flux and performing two T-dualities leads to non-commutativity in the coordinates of the system. It was known before for the closed sting that an additional T-duality leads to a non-associative phase space. Here, we extend the result to open strings and see the same effect on the phase space of the open string endpoint: Using symmetries we are able to show that the phase space becomes non-associative.

The chapter is organized as follows: We first review the commutation relations for the closed string in the Q-flux backgrounds as well as in the R-flux background. Then, we introduce the Seiberg-Witten map for the open string which gives us an description without a B-field but with a non-commutative spacetime. Afterwards, we compute the commutation relations for the open string in geometric and non-geometric background fields. The computations are based on the paper [26].

9.1 String Non-Associativity for Closed Strings

In the papers [93, 94, 95, 96, 97, 98] it was shown that in certain NS-NS background configurations the behavior of closed strings is non-commutative or even non-associative.

Starting from a closed string on a three-dimensional torus with H-flux, the commutation relations for the closed string coordinates have been computed in a world-sheet calculation. For the Q-flux background, it was shown in [94, 99, 100, 101, 102] that the commutator between two-closed string coordinates depends on the winding number as

$$\left[X^i, X^j\right]_{\text{closed}} = iQ_k{}^{ij}n^k\,. \tag{9.1}$$

As a consequence of the non-locality of the winding number n^k, the closed string non-commutativity is a non-local effect and is only consistent in string theory. Its non-locality arises from the way the Q-flux is defined: it can be viewed as a combination of metric and B-field background which is globally patched in a consistent way by a T-duality.

Going the duality chain 7.3.1 one step further, we find the non-geometric R-flux background. This non-geometric flux is proportional to the Jacobinator, which measures the violation of the Jacobi-identity and is therefore also a measure of the non-associativity. The commutation relations in the R-flux background are given by

$$\Big[X^i, X^j\Big]_{\text{closed}} = i\, R^{ijk}\, p_k \,. \tag{9.2}$$

Here, p_k is the momentum operator. This can be interpreted that the R-flux background is not even locally a well-defined manifold and needs a T-duality transformation at every point to be correctly patched. Using the non-vanishing canonical commutation relations of position and momentum operator, we find a non-vanishing three-bracket and therefore a non-associative behavior given by

$$\begin{aligned} &\Big[X^k, p_k\Big] = i\,, \\ &\Big[X^i, X^j, X^j\Big]_{\text{closed}} := \Big[[X^i, X^j], X^j\Big]_{\text{closed}} \pm \text{perm.} \simeq R^{ijk}\,. \end{aligned} \tag{9.3}$$

Such a structure in the phase space is known as *twisted Poisson structure*. A similar result was derived in [103] by quantization an associated membrane sigma model. For a more mathematical discussion, we refer to [104, 105, 106, 107] where fibrations were used to characterize non-geometric backgrounds with D-branes and B-fields.

Note that the closed-string coordinates X^i and their T-dual coordinates $\tilde{X}_i$ are not analogous to position and momentum and therefore do not obey canonical commutation relation of this form. Instead their commutator vanishes

$$\Big[X^i, \tilde{X}_j\Big]_{\text{closed}} = 0 \neq i\,\delta^i{}_j \,. \tag{9.4}$$

For deeper insights, we refer to the papers [108, 109].

9.2 Open Strings in B-Field Backgrounds

We find a similar result for the open string. For this computation, we introduce the Seiberg-Witten map which enables us to map an open string with a B-field to an open string without a B-field but within a non-commutative spacetime. Seiberg and Witten studied the case for a constant B-field first, and later it was generalized to the case for a non-constant B-field. We will briefly mention their arguments for the case with constant B-field. For the non-constant B-field, we refer to the argumentation in [110, 111].

For the constant B-field we start with the world-sheet action

$$S = \frac{1}{4\pi\alpha'}\int_\Sigma d^2\sigma\, g_{ij}\,\partial_a \mathsf{X}^i \partial^a \mathsf{X}^j - \frac{i}{4\pi\alpha'}\int_{\partial\Sigma} d\sigma\, B_{ij}\mathsf{X}^i \partial_t \mathsf{X}^j \,, \tag{9.5}$$

where Σ is the string world-sheet with Euclidean signature, $\partial\Sigma$ denotes the corresponding boundary with the top-form $d\sigma$, and $d^2\sigma$ denotes the top-form on Σ. As a consequence of the B-field the boundary conditions of the X^i-direction on the p-brane change to

$$g_{ij}\partial_n \mathsf{X}^j + B_{ij}\partial_t \mathsf{X}^j\Big|_{\partial\Sigma} = 0\,, \tag{9.6}$$

where ∂_n is the derivative orthogonal to the boundary $\partial\Sigma$. Using the conformal map and introducing the variables z and $\bar{z}$, we end up in the upper half plane with the boundary condition

$$g_{ij}\left(\partial - \bar{\partial}\right)\mathsf{X}^j + B_{ij}\left(\partial + \bar{\partial}\right)\mathsf{X}^j\Big|_{\partial\Sigma} = 0\,, \tag{9.7}$$

where $\partial = \frac{\partial}{\partial z}$ and $\bar{\partial} = \frac{\partial}{\partial \bar{z}}$. The boundary of the string $\partial\Sigma$ is mapped to the real line. The propagator of the string with these boundary conditions is [112, 113, 114]

$$\begin{aligned}\left\langle \mathsf{X}^i(z)\mathsf{X}^j(z')\right\rangle = &- \alpha'\Big(g^{ij}\log|z-z'| - g^{ij}\log|z-\bar{z}'| \\ &+ G^{ij}\log|z-\bar{z}'|^2 + \theta^{ij}\log\frac{z-\bar{z}'}{\bar{z}-z'} + D^{ij}\Big)\,,\end{aligned} \tag{9.8}$$

where

$$\begin{aligned} G_{ij} &= \left[g - Bg^{-1}B\right]_{ij}\,, \\ G^{ij} &= \left[(g+B)^{-1}\right]^{ij}_{S} = +\left[(g+B)^{-1}g(g-B)^{-1}\right]^{ij}\,, \\ \theta^{ij} &= \left[(g+B)^{-1}\right]^{ij}_{A} = -\left[(g+B)^{-1}B(g-B)^{-1}\right]^{ij}\,. \end{aligned} \tag{9.9}$$

Here, $()_A$ and $()_S$ denote the anti-symmetric and the symmetric part of the matrix. The constants D^{ij} in (9.8) can be neglected since they are independent of z or z'.

For the purpose of our later analysis, we restrict ourselves to the boundary points of the string with $z = \tau$ and $z' = \tau'$. A careful computation shows that for the boundary points we end up with

$$\left\langle \mathsf{X}^i(\tau)\,\mathsf{X}^j(\tau')\right\rangle = -\alpha' G^{ij}\log(\tau-\tau')^2 + i\alpha'\pi\theta^{ij}\epsilon(\tau-\tau')\,. \tag{9.10}$$

Therefore, the commutator for the open string endpoints becomes

$$\left[\mathsf{X}^i(\tau),\mathsf{X}^j(\tau)\right] = T\left(\mathsf{X}^i(\tau)\,\mathsf{X}^j\left(\tau^-\right) - \mathsf{X}^i(\tau)\,\mathsf{X}^j\left(\tau^+\right)\right) = 2\pi\alpha' i\theta^{ij}\,. \tag{9.11}$$

As we can see from this equation, the anti-symmetric quantity θ measures the anti-commutativity, whereas G^{ij} is the metric on the brane of the endpoints of the open strings.

Therefore, we can summarize the map from an open string with B-field B_{ij} to an open string in an anti-commutative spacetime with parameter θ^{ij} by

$$(g+B)^{-1} = G + \theta \tag{9.12}$$

As next step, we are going to compute the commutation relation for the open string endpoints of an open string with Neumann-Neumann boundary condition $X_{NN}\left(z,\bar{z}\right)$ and an open string with the Dirichlet-Dirichlet boundary condition $X_{DD}\left(z,\bar{z}\right)$. Note that these two strings are T-dual to each other. As a convention, we are going to denote the Neumann-Neumann string as open string $X\left(z,\bar{z}\right)=X_{NN}\left(z,\bar{z}\right)$, and the Dirichlet-Dirichlet string as the T-dual one $\tilde{X}\left(z,\bar{z}\right)=X_{DD}\left(z,\bar{z}\right)$. The mode expansions are

$$\begin{aligned} X(z,\bar{z}) &= q - i\alpha' p \log|z|^2 + i\sqrt{\frac{\alpha'}{2}}\sum_{n\neq 0}\frac{1}{n}\alpha_n(z^{-n}+\bar{z}^{-n})\,,\\ \tilde{\mathsf{X}}(z,\bar{z}) &= q_0 + \frac{1}{2\pi i}(q_1-q_0)\log\left(\frac{z}{\bar{z}}\right) + i\sqrt{\frac{\alpha'}{2}}\sum_{n\neq 0}\frac{1}{n}\alpha_n(z^{-n}-\bar{z}^{-n})\,, \end{aligned} \tag{9.13}$$

with q and p the center of mass position and momentum for the Neumann-Neumann string, and q_0 and q_1 the endpoints of the open strings on the branes. The two-point function can be directly computed and is

$$\langle \mathsf{X}(z,\bar{z})\tilde{\mathsf{X}}(w,\bar{w})\rangle = -\frac{\alpha'}{2}\left(\log\left(\frac{z-w}{\bar{z}-\bar{w}}\right)+\log\left(\frac{\bar{z}-w}{z-\bar{w}}\right)\right)\,. \tag{9.14}$$

Using the Seiberg-Witten notation one can directly deduce that the computation relation for the open string and its dual is

$$\left[\mathsf{X}(\tau),\tilde{\mathsf{X}}(\tau)\right] = \pi\alpha' i\,. \tag{9.15}$$

Here, we can already see a qualitative difference between open and closed string: Whereas the equal-time commutator vanishes for the closed string and its dual, it is non-zero for the open string.

9.3 Open Strings on a Three-Dimensional Torus with Fluxes

In this section, we are going to demonstrate the behavior of a closed and an open string in a non-geometric background. Therefore, we are using the T-duality chain:

$$H_{ijk} \quad \xrightarrow{T_k} \quad F_{ij}{}^{k} \quad \xrightarrow{T_j} \quad Q_i{}^{jk} \quad \xrightarrow{T_i} \quad R^{ijk}\,. \tag{9.16}$$

These strings are only toy-examples for strings. In physical examples one would have to add additional fluxes, and not just have one at a time.

For the open string-computation we have to choose a configuration for the D-branes. We are going to start from a point-like D0-brane with an H-flux. Performing a T-duality transformation perpendicular to an arbitrary brane will raise the dimensionality of the brane, whereas a T-duality in the direction of the brane will lower the dimension. Therefore, we will end with a space-time filling D3-brane on the T^3-torus. For the world-sheet computation which are necessary to find the results, we refer to the Appendix B and to the original paper [26].

9.3.1 T^3 with H-Flux and D0-branes

As background we have a three-torus $\mathbb{T}^3$ with a non-vanishing H-flux with values $H_{123} = N$. Note that the B-field depends on the closed string coordinate $X^3_{\rm cl}$:

$$B_{12} = N X^3_{\rm cl} \tag{9.17}$$

In this setting we start with a single point-like D0-brane, and therefore, we are going to have only strings with Dirichlet-Dirichlet boundary conditions. The commutator for the open string endpoints vanishes,

$$\left[\tilde{\mathsf{X}}_i, \tilde{\mathsf{X}}_j\right] \equiv [\mathsf{X}_{i\,\rm DD}, \mathsf{X}_{j\,\rm DD}] = 0\,, \qquad\qquad i,j = 1,2,3\,, \tag{9.18}$$

and therefore commute with each other.

9.3.2 Twisted T^3 and D1-branes

As next step, we perform a T-duality transformation along the x^1-direction and find a twisted torus as background. As a result of the T-duality in the X^1-direction the D0-brane turns into a D1-brane in the x^1-direction. One consequence of having a geometric flux is that the geometry of the torus is twisted and therefore the D1-brane has an angle depending on the value of the geometric flux (see for example [79]). The open string coordinates commute in each direction, i.e.

$$\begin{aligned} \left[\mathsf{X}^1, \tilde{\mathsf{X}}_i\right] &\equiv \left[\mathsf{X}^1_{\rm NN}, \mathsf{X}_{i\,\rm DD}\right] &&= 0 \qquad \text{with} \qquad i = 2,3\,, \\ \left[\tilde{\mathsf{X}}_2, \tilde{\mathsf{X}}_3\right] &\equiv \left[\mathsf{X}_{2\,\rm DD}, \mathsf{X}_{3\,\rm DD}\right] &&= 0\,. \end{aligned} \tag{9.19}$$

9.3.3 $\mathbb{T}^3$ with Q-Flux and D2-branes

We perform a T-duality transformation along the x^2-direction to end up with a D2-brane in a T-fold in the directions x^1 and z^2. The background has a non-geometric Q-flux with value $Q = N$.

We can apply the Seiberg-Witten mapping (9.9) to this configuration and find for the non-commutative space the following metric and the non-commutativity parameter:

$$\begin{aligned} d\hat{s}^2 =& r_1^2\left(dX^1\right)^2 + r_2^2\left(dX^2\right)^2 + \frac{1}{r_3^2}\left(dX^3\right)^2\,, \\ \theta^{12} =& - N X^3_{cl}\,. \end{aligned} \tag{9.20}$$

Therefore, we find for the endpoints of the open string a non-zero commutator

$$\left[\mathsf{X}^1, \mathsf{X}^2\right] \equiv \left[\mathsf{X}^1_{\rm NN}, \mathsf{X}^2_{\rm NN}\right] = 2\pi\alpha' i \theta^{12} = -2\pi\alpha' i Q X^3_{cl}\,. \tag{9.21}$$

This is analogue to the closed-string commutation relation in the Q-flux setting described in (9.1). The difference is that we do not have a winding mode proportional to the commutator but to the closed string coordinate of the background.

9.3.4 $\mathbb{T}^3$ with R-Flux and D3-branes

Finally, we perform a T-duality transformation along the x^3-direction on the torus. As we have argued in section 7.3.1, this is in double field theory allowed and thus we perform such transformation formally. Therefore, the closed string coordinate X^3 becomes the dual coordinate $\tilde{X}^3$ and we find the expected R-flux in this situation.

Thus, the commutation relations for the open-string endpoints become

$$\left[\mathsf{X}^1, \mathsf{X}^2\right] \equiv \left[\mathsf{X}^1_{\mathrm{NN}}, \mathsf{X}^2_{\mathrm{NN}}\right] = -2\pi\alpha' i R \tilde{X}_{cl,3} \,. \tag{9.22}$$

To show the equivalence between open and closed string sector we have to calculate the three-bracket for the open strings in the R-flux background. Evaluating the three bracket we find that

$$\left[\mathsf{X}^1_{\mathrm{NN}}, \mathsf{X}^2_{\mathrm{NN}}, \mathsf{X}^3_{\mathrm{NN}}\right] = -2\pi\alpha' i R \left[\tilde{X}_{cl,3}, \mathsf{X}^3_{\mathrm{NN}}\right] . \tag{9.23}$$

As final step we have to compute the commutator between an open-string coordinate and the dual closed-string coordinate. The mode expansion for the dual closed string is

$$\tilde{X}_{\mathrm{closed}}(z,\bar{z}) = q_0 + \frac{p_0}{2\pi i}\log\frac{z}{\bar{z}} + i\sqrt{\frac{\alpha'}{2}}\sum_{n\neq 0}\frac{1}{n}\left(\alpha_n z^{-n} - \bar{\alpha}_n \bar{z}^{-n}\right) . \tag{9.24}$$

Then, the commutator between the open and the dual closed coordinate becomes

$$\left[\mathsf{X}_{\mathrm{NN}}(\tau,\sigma), \tilde{X}_{\mathrm{closed}}(\tau',\sigma')\right] = \frac{\alpha'}{2}\sum_{n\neq 0}\frac{1}{n}\left(z^{-n} + \bar{z}^{-n}\right)\left(z'^{n} - \bar{z}'^{n}\right) . \tag{9.25}$$

Evaluating the infinite sum at the endpoint of the string we find that

$$\left[\mathsf{X}_{\mathrm{NN}}(\tau,0), \tilde{X}_{\mathrm{closed}}(\tau',0)\right] = \begin{cases} +i\pi\alpha' & \text{if } \tau > 0\,, \\ -i\pi\alpha' & \text{if } \tau < 0\,. \end{cases} \tag{9.26}$$

Using this result for equation (9.23), we find for the three-bracket the same result as for the closed string case, namely

$$\left[\mathsf{X}^1_{\mathrm{NN}}, \mathsf{X}^2_{\mathrm{NN}}, \mathsf{X}^3_{\mathrm{NN}}\right] = -2\pi\alpha' i R \left[\tilde{X}_3, \mathsf{X}^3_{\mathrm{NN}}\right] = -2\pi^2\alpha'^2 R \,. \tag{9.27}$$

Therefore, we have been able to successfully find the commutation relation for the open string coordiantes in the R-flux background. As expected the Jacobinator is proportional to the R-flux and therefore not vanishing in non-geometric backgrounds.

Part IV

Conclusions

Chapter 10

Summary and Outlook

Summary

After an introduction in part I, we started part II on the basic ideas of machine learning (ML) and presented here the main concepts such as the idea of neural networks and back propagation in chapter 3. Additionally, we presented a convolutional neural network, one of the prime examples for the recent success of machine learning, to give an insight on the simplicity of those methods allowing simple approaches for complicated problems.

In the next chapters, we presented our machine learning based methods to discover different types of symmetries. We hope that these tools will allow others to understand their problem settings better and help them to discover underlying symmetries and dualities and allow them to answer their open questions.

In chapter 4, we showed how one can use the representation in hidden layers to find symmetries in a dataset. For this, we trained the neural network on a classification task. The training process clusters points linked by a symmetry in the hidden layers of the neural network which enables us to find such classes within our dataset. We used these clusters in a second step to identify the underlying symmetry group by an automated algorithm. The true strength of this method is the broad scope of the method: it can be applied to different kinds of problem settings to investigate datasets without any previous understanding of the structure, and therefore allows us to get a first understanding of a complicated physical question.

In chapter 5, we focused on learning conserved quantities of classical physical systems. We modified the described Hamiltonian Neural Network and extended it to the Symmetry Control Neural Network. For the Symmetry Control Neural Network, we combined the idea of coordinate transformations with the idea of the Hamiltonian Neural Network. This allowed us to find a coordinate transformation to a coordinate system with conserved quantities as generalized momenta. Reading out those conserved quantities allowed us to get analytical formulae for the conserved quantities. We also suggested a modification which would allow us to learn conserved quantities in physical systems with known Hamiltonian.

Finally, in chapter 6, we presented a method how to learn Lax pairs, respectively Lax connection of physical systems. As in the chapter before, we constructed a suitable loss function to force the neural network to find solutions to the Lax pair condition. In the second part, we focused on the integrability of perturbations: Knowing a suitable Lax pair for the unperturbed system we developed a method to check whether a perturbation is integrable or not.

The methods of chapters 5 and 6 can be seen as two examples of the idea to constrain the output of the neural network by physical equations. The neural network is going to search for solutions to fulfill the given constraints and allowing us to tackle highly complicated problem settings.

In contrast to this is the third part of this thesis focused on the exploitation of already known dualities in string theories. Chapter 7 begins with a short repetition of the different dualities and their underlying symmetry groups in string theory, followed by an introduction to exceptional field theory, the generalized diffeomorphisms as well as the transformation behavior of the fields and the coordinates under U-dualities.

In the following two chapters we examined the effects of those U-dualities on twisted tori and the consequences of such geometries on the phase space. In chapter 8, we constructed the complete flux content of exceptional field theories for the dimensions $d = 4, 5, 6, 7$. In the second step, we presented duality chains to examine the impact of non-geometric backgrounds. Forbidden wrapping modes for the twisted tori, the starting point of the duality chains, resulted in missing momenta modes within non-geometric backgrounds – for the most „non-geomtric“ flux no momenta modes are allowed.

In the last chapter, we were able to construct a duality chain which reveals non-associativity for the end-points of open strings in the locally non-geometric space. This non-vanishing Jacobinator leads to a deformation of the phase-space.

Outlook

In this thesis, we presented several ways how to find symmetries using neural network. We hope that these methods will be utilized in the future to detect unknown symmetries in physical problem settings extending our understanding of the phenomena in our world and hopefully will lead to a unified conception of our universe.

Along this course of this thesis, we presented several ways to implement a functional bias to neural networks. The idea to constrain the functions of the neural network using equations from the theoretical understanding of the problem. This can be used to improve previous results of machine learning, as well as to apply machine Learning to additional areas of physics using machine learning as a tool to find new structures hidden in the physical theory.

Here, we will collect some ideas for applications for the presented tools in this thesis. In general, there are many possible applications of machine learning in theoretical physics, especially in string theory. The challenge for many cases is to find the right question and frame the problem in the right way to make it accessible with machine learning. Then, machine learning is able to provide necessary tools to improve the understanding of problems in string theory and theoretical physics in general.

- An interesting research direction is the understanding of the clustering in chapter 4 for the Complete Intersection Calabi-Yau manifolds. The observation that the neural network is able to “identify” classes of the dataset depending on topological invariants it is not aware of leads to the conclusion that the neural network has some kind of understanding how to find a suitable basis to compute those invariants. Such a formula is yet unknown but would yield a deeper understanding of the Complete Intersection Calabi-Yau manifolds.

- The methodology to find a Lax-Pair has not been applied to systems where a Lax pair is not known yet. The given framework can be used to search for integrable deformations in sigma models, as it is presented in [115, 116] or to search for integrable sectors in the Ω deformation. To extend this framework to quantum integrability, one has to include the Yang-Baxter equations in the formalism, but this seems fairly accessible by defining a suitable loss function.

- We already mentioned the idea to use reinforcement learning to examine the string landscape. In [33], the author of this thesis, along with others, looked for stable string vacua in the type IIB flux landscape. Extending this approach to spaces with more moduli might be an approach to find suitable string theory vacua to solve the problems arising from KKLT [117]. Here, reinforcement learning is a suitable tool to examine the space of possible string vacua and help to improve the understanding of the structure of string vacua.

- Further applications of the Symmetry Control Neural Network do not lie necessarily in the context of mathematical physics, but might be applicable to many problems for simulations in Machine Learning. An interesting application is the current development to use Hamiltonian flows for a faster sampling of $\mathrm{SU}(N)$ gauge theories [118]. Here, the idea of Symmetry Control Neural Network should help to improve the results, speed up the sampling process and make it more stable.

- It would be interesting to understand the non-geometric fluxes found in 8 from the string theory perspective as R-R fluxes. In [88] the fluxes of the SL(5) exceptional field theory have been understood from the Type IIB perspective. A similar analysis for the previously unknown non-geometric R-fluxes should mix the R-R sector with the NS-NS sector in a similar fashion.

- The results of chpater 9 are currently only for the perturbative regime of a low flux density which is a appropriate approach when choosing suitable values for the radii of the three-torus. For a proper string-background it would be interesting to check the non-associativity also in the non-perturbative regime as it was already done for the closed string [102].

Part V

Appendix

Appendix A

Mathematical details of Exceptional Field Theory

In this part of the appendix, we present the details for the different dimensions of exceptional field theory which are used in the chapters 7 and 8.

A.1 Details of Decomposition of $\mathrm{E}_{d(d)}$ EFT to $\mathrm{SL}(d)\times\mathbb{R}^+$

A.1.1 $\mathrm{SL}(5)$ EFT

The generalized diffeomorphism acting on the generalized vielbein $E_{\bar{M}}{}^M$ is given by

$$\begin{aligned}\mathcal{L}_V E_{\bar{M}}{}^M &= V^N\partial_N E_{\bar{M}}{}^M - V^N\partial_N\Lambda^M + Y^{MP}_{NQ}V^N\partial_P\Lambda^Q + \omega_d V^M\partial_N\Lambda^N\\ &= V^N\partial_N E_{\bar{M}}{}^M - E_{\bar{M}}{}^N\partial_N V^M + \eta^{\alpha MP}\eta_{\alpha NQ}E_{\bar{M}}{}^N\partial_P V^Q - \frac{1}{5}E_{\bar{M}}{}^M\partial_N V^N\,,\end{aligned} \tag{A.1}$$

where $M, N = 1, \ldots, 10$ labels the spinor and $\alpha = 1, \ldots, 5$ the fundamental representation of SL(5), $Y^{MP}_{NQ} = \eta^{\alpha MP}\eta_{\alpha NQ}$ is the Y-tensor defined in section 7.2. Here, the weight of the vielbein is already set to 0. Note that we have used this generalized diffeomorphism to calculate the exact structure of the generalized vielbein demanding that

$$E^{\bar{M}}{}_M\mathcal{L}_V E_{\bar{M}}{}^M = \mathcal{L}_V|E| = 0\,, \tag{A.2}$$

and as a consequence we can take $E_{\bar{M}}{}^M$ to have unit determinant. The section condition is

$$\eta^{\alpha MN}\partial_M\otimes\partial_N = \eta^{\alpha MN}\partial_M\partial_N = 0\,. \tag{A.3}$$

Using the decomposition $\mathrm{SL}(5)\longrightarrow\mathrm{SL}(4)\times\mathbb{R}^+$, generalized vectors in the **10** and **5** of SL(5) decompose as follows

$$V^M = \left(V^i,\, V_{ij}\right)\,,\qquad V^\alpha = \left(V^i,\, V^5\right)\,. \tag{A.4}$$

In the SL(4) × $\mathbb{R}^+$ basis, we find the following η-matrices to be non-zero:

$$\begin{aligned} \eta^{\alpha\, MN}: \quad & \eta^{ij}_{kl} = 2\delta^{ij}_{kl}\,, \qquad \eta_{i\,jk\,lm} = \frac{1}{\sqrt{2}}\epsilon_{ijklm}\,, \\ \eta_{\alpha\, MN}: \quad & eta^{kl}_{ij} = 2\delta^{kl}_{ij}\,, \qquad \eta^{i\,jk\,lm} = \frac{1}{\sqrt{2}}\epsilon^{ijklm}\,. \end{aligned} \tag{A.5}$$

Note that the η-matrices are symmetric in the M, N indices, i.e. $\eta^{\alpha\, MN} = \eta^{\alpha\, NM}$. Using (A.5), we obtain the section condition in $d = 4$, which has been stated in (8.15). Furthermore, using (A.5) and (A.1) we find as transformation behavior of the vielbein under the generalized Lie derivative corresponding to a spacetime diffeomorphism $V^M = (\xi^i, 0, 0)$ the following relations

$$\begin{aligned} \mathcal{L}_\xi E_{\bar M}{}^i &= \xi^j \partial_j E_{\bar M}{}^i - E_{\bar M}{}^j \partial_j \xi^i - 2E_{\bar M\, jk}\partial^{ji}\xi^k - \frac{2}{5}E_{\bar M}{}^i \partial_j \xi^j\,, \\ \mathcal{L}_\xi E_{\bar M\, ij} &= \xi^k \partial_k E_{\bar M\, ij} - 2E_{\bar M k[i}\partial_{j]}\xi^k + \frac{3}{5}E_{\bar M\, ij}\partial_k \xi^k\,, \end{aligned} \tag{A.6}$$

From this, we can find the correct weights of the columns of the vielbein. Additionally, we can easily recover the transformation law for Ω^{ijk} given in equation (8.9).

A.1.2 SO(5, 5) EFT

The generalised diffeomorphism acting on the generalised vielbein $E_{\bar M}{}^M$ is given by [119, 68]

$$\begin{aligned} \mathcal{L}_V E_{\bar M}{}^M &= V^N \partial_N E_{\bar M}{}^M - V^N \partial_N \Lambda^M + Y^{MP}_{NQ} V^N \partial_P \Lambda^Q + \omega_d V^M \partial_N \Lambda^N \\ &= V^N \partial_N E_{\bar M}{}^M - E_{\bar M}{}^N \partial_N V^M + \frac{1}{2}\left(\gamma^I\right)^{MP} \left(\gamma_I\right)_{NQ} E_{\bar M}{}^N \partial_P V^Q - \frac{1}{4} E_{\bar M}{}^M \partial_N V^N\,, \end{aligned} \tag{A.7}$$

where $M, N = 1, \ldots, 16$ labels the spinor and $I = 1, \ldots, 10$ the fundamental representation of SO(5, 5), $Y^{MP}_{NQ} = \frac{1}{2}\left(\gamma^I\right)^{MP}\left(\gamma_I\right)_{NQ}$ is Y-tensor product defined in section 7.2. Here, the weight of the vielbein is already set to 0. Here $\left(\gamma_I\right)^{MN}$ are SO(5, 5) γ-matrices, satisfying

$$\left(\gamma_{(I}\right)^{MP}\left(\gamma_{J)}\right)_{NP} = \eta_{IJ}\delta^M_N\,, \tag{A.8}$$

where η_{IJ} is the SO(5, 5) invariant metric. Note that we have used the generalized diffeomorphism to calculate the exact structure of the generalized vielbein demanding that

$$E^{\bar M}{}_M \mathcal{L}_V E_{\bar M}{}^M = \mathcal{L}_V |E| = 0\,, \tag{A.9}$$

and as a consequence we can define $E_{\bar M}{}^M$ in such a way that it has unit determinant. The section condition is

$$\left(\gamma_I\right)^{MN} \partial_M \otimes \partial_N = \left(\gamma_I\right)^{MN} \partial_M \partial_N = 0\,. \tag{A.10}$$

Using the decomposition $\mathrm{SO}(5,5) \longrightarrow \mathrm{SL}(5) \times \mathbb{R}^+$, generalized vectors in the **16** and **10** of $\mathrm{SO}(5,5)$ decompose as

$$V^M = \left(V^i,\, V_{ij},\, V_z\right) \,, \qquad V^I = \left(V^i,\, V_i\right) \,. \tag{A.11}$$

In the $\mathrm{SL}(5) \times \mathbb{R}^+$ basis, we find the following γ-matrices to be non-zero:

$$\begin{aligned} (\gamma_I)^{MN} : &\quad \left(\gamma^i\right)^j{}_{kl} = 2\delta^{ij}_{kl} \,, \qquad (\gamma_i)^j{}_z = \sqrt{2}\delta^j_i \,, \qquad (\gamma_i)_{jk\,lm} = \frac{1}{\sqrt{2}}\epsilon_{ijklm} \,, \\ (\gamma_I)_{MN} : &\quad (\gamma_i)_j{}^{kl} = 2\delta^{kl}_{ij} \,, \qquad \left(\gamma^i\right)_j{}^z = \sqrt{2}\delta^i_j \,, \qquad \left(\gamma^i\right)^{jk\,lm} = \frac{1}{\sqrt{2}}\epsilon^{ijklm} \,. \end{aligned} \tag{A.12}$$

Note that the γ-matrices are symmetric in the indices M, N, i.e. $(\gamma_I)^{MN} = (\gamma_I)^{NM}$. Using (A.12), we obtain the section condition in $d = 5$, which have been stated in (8.36). Furthermore, using (A.12) and (A.7) we find as transformation behavior under the generalized Lie derivative corresponding to a spacetime diffeomorphism $V^M = (\xi^i, 0, 0)$ the following relations

$$\begin{aligned} \mathcal{L}_\xi E_{\bar{M}}{}^i &= \xi^j \partial_j E_{\bar{M}}{}^i - E_{\bar{M}}{}^M \partial_M \xi^i - 2E_{\bar{M}\,jk}\partial^{ji}\xi^k + E_{\bar{M}\,z}\partial^z \xi^i - \frac{1}{4}E_{\bar{M}}{}^i \partial_j \xi^j \,, \\ \mathcal{L}_\xi E_{\bar{M}\,ij} &= \xi^k \partial_k E_{\bar{M}\,ij} - 2E_{\bar{M}k[i}\partial_{j]}\xi^k + \frac{1}{2}\epsilon_{ijklm}E_{\bar{M}\,z}\partial^{kl}\xi^m - \frac{1}{4}E_{\bar{M}\,ij}\partial_k \xi^k \,, \\ \mathcal{L}_\xi E_{\bar{M}\,z} &= \xi^i \partial_i E_{\bar{M}\,z} + \frac{3}{4}E_{\bar{M}\,z}\partial_k \xi^k \,. \end{aligned} \tag{A.13}$$

From this, we can find the correct weights of the columns of the vielbein. Additionally, we can easily recover the transformation law for Ω^{ijk} given in equation (8.35).

A.1.3 $\mathrm{E}_{6(6)}$ EFT

The generalized diffeomorphism acting on the generalized vielbein $E_{\bar{M}}{}^M$ is given by

$$\begin{aligned} \mathcal{L}_V E_{\bar{M}}{}^M &= V^N \partial_N E_{\bar{M}}{}^M - V^N \partial_N \Lambda^M + Y^{MP}_{NQ} V^N \partial_P \Lambda^Q + \omega_d V^M \partial_N \Lambda^N \\ &= V^N \partial_N E_{\bar{M}}{}^M - E_{\bar{M}}{}^N \partial_N V^M + 10 d^{MNP} d_{PQR} E_{\bar{M}}{}^Q \partial_N V^R - \frac{1}{3}E_{\bar{M}}{}^M \partial_N V^N \,, \end{aligned} \tag{A.14}$$

where $M, N = 1, \dots, 27$ label the fundamental representation of $\mathrm{E}_{6(6)}$,
$Y^{MP}_{NQ} = 10\ d^{MNP}\ d_{PQR}$ is Y-tensor product defined in section 7.2 and d_{MNP} and d^{MNP} are the symmetric cubic $\mathrm{E}_{6(6)}$ invariant tensors. Here, the weight of the vielbein is already set to 0. Here $(\gamma_I)^{MN}$ are $\mathrm{SO}(5,5)$ γ-matrices, satisfying

$$d^{MPQ} d_{NPQ} = \delta^M_N \,, \tag{A.15}$$

where η_{IJ} is the $\mathrm{SO}(5,5)$ invariant metric. Note that we have used this generalized diffeomorphism to calculate the exact structure of the generalized vielbein demanding that

$$E^{\bar{M}}{}_M \mathcal{L}_V E_{\bar{M}}{}^M = \mathcal{L}_V |E| = 0 \,, \tag{A.16}$$

and as a consequence we can take ${E_{\bar{M}}}^{M}$ to have unit determinant. The section condition is

$$d^{MNP}\partial_M \otimes \partial_N = d^{MNP}\partial_M \partial_N = 0\,, \tag{A.17}$$

Using the decomposition $\mathrm{E}_{6(6)} \longrightarrow \mathrm{SL}(6)\times\mathrm{SL}(2) \longrightarrow \mathrm{SL}(6)\times\mathbb{R}^+$, generalized vectors of $\mathrm{E}_{6(6)}$ decompose as

$$V^M = \left(V^i,\, V_{ij},\, V_{ijklm}\right)\,. \tag{A.18}$$

The only non-zero components of cubic invariants are given by

$$\begin{aligned}
d^{MNP}:\quad & {d^i}_{jk\,lmnpq} = \frac{1}{10\sqrt{6}}\delta^i_{[j}\epsilon_{k]lmnpq}\,, \qquad && d_{ij\,kl\,mn} = \frac{1}{4\sqrt{5}}\epsilon_{ijklm}\,,\\
d_{MNP}:\quad & d_i^{\;jk\,lmnpq} = \frac{1}{10\sqrt{6}}\delta_i^{[j}\epsilon^{k]lmnpq}\,, \qquad && d^{ij\,kl\,mn} = \frac{1}{4\sqrt{5}}\epsilon^{ijklmn}\,.
\end{aligned} \tag{A.19}$$

Using (A.19), we obtain the section condition in $d = 6$, which have been stated in (8.53). Furthermore, using (A.19) and (A.14) we find as transformation behavior under the generalized Lie derivative corresponding to a spacetime diffeomorphism $V^M = (\xi^i, 0, 0)$ the relations

$$\begin{aligned}
\mathcal{L}_\xi {E_{\bar{M}}}^i =& \xi^j\partial_j {E_{\bar{M}}}^i - {E_{\bar{M}}}^j\partial_j\xi^i - \frac{6}{5!}E_{\bar{M}\,j_1j_2j_3j_4j_5}\partial^{[ij_1j_2j_3j_4}\xi^{j_5]}\\
&- 3E_{\bar{M}\,jk}\partial^{[jk}\xi^{i]} - \frac{1}{3}{E_{\bar{M}}}^i\partial_j\xi^j\,,\\
\mathcal{L}_\xi E_{\bar{M}\,ij} =& \xi^k\partial_k E_{\bar{M}\,ij} + 2E_{\bar{M}\,k[j}\partial_{i]}\xi^k + E_{\bar{M}\,ik}\partial_j\xi^k\\
&- \sqrt{30}E_{\bar{M}\,ijklm}\partial^{kl}\xi^m - \frac{1}{3}E_{\bar{M}\,ij}\partial_k\xi^k\,,\\
\mathcal{L}_\xi E_{\bar{M}\,i_1i_2i_3i_4i_5} =& \xi^j\partial_j E_{\bar{M}\,i_1i_2i_3i_4i_5} + 5E_{\bar{M}\,j[i_1i_2i_3i_4}\partial_{i_5]}\xi^j - \frac{1}{3}E_{\bar{M}\,i_1i_2i_3i_4i_5}\partial_j\xi^j\,.
\end{aligned} \tag{A.20}$$

From this, we can find the correct weights of the columns of the vielbein. Additionally, we can easily recover the transformation law for Ω^{ijk} and for Ω^{ijklmn} given in equation (8.50).

A.1.4 $\mathrm{E}_{7(7)}$ EFT

Let us conclude with the $\mathrm{E}_{7(7)}$ exceptional field theory. For the generalized Lie derivative of the generalized vielbein ${E_{\bar{M}}}^M$ we find

$$\begin{aligned}
\mathcal{L}_V {E_{\bar{M}}}^M &= V^N\partial_N {E_{\bar{M}}}^M - V^N\partial_N\Lambda^M + Y^{MP}_{NQ}V^N\partial_P\Lambda^Q + \omega_d V^M\partial_N\Lambda^N\\
&= V^N\partial_N {E_{\bar{M}}}^M - {E_{\bar{M}}}^N\partial_N V^M - 12\,(t_\alpha)^{MP}\,(t^\alpha)_{NQ}\,{E_{\bar{M}}}^N\partial_P V^Q\\
&\quad + \frac{1}{2}\Omega^{MP}\Omega_{NQ}{E_{\bar{M}}}^N\partial_P V^Q - \frac{1}{2}{E_{\bar{M}}}^M\partial_N V^N\,,
\end{aligned} \tag{A.21}$$

where $M, N = 1, \ldots, 56$ label the fundamental and $\alpha, \beta = 1, \ldots, 133$ label the adjoint representations of $\mathrm{E}_{7(7)}$.

$$Y^{MP}_{NQ} = \frac{1}{24}\delta^M_N\delta^P_Q + \frac{1}{12}\delta^M_Q\delta^P_Q + (t_\alpha)^{MP}\,(t^\alpha)_{NQ} - \frac{1}{24}\Omega^{MP}\Omega_{NQ}\,, \tag{A.22}$$

is the Y-tensor of $\mathrm{E}_{7(7)}$, with $(t_\alpha)^{MN}$ the generators of $\mathrm{E}_{7(7)}$ in the fundamental representations. For raising and lowering of the indices $M, N, \ldots$, we use the symplectic invariant Ω_{MN} of $\mathrm{Sp}(56) \supset \mathrm{E}_{7(7)}$, which we defined using the north-west south-east convention, i.e.

$$V^M = \Omega^{MN} V_N \,, \qquad V_M = V^N \Omega_{MN} \,, \tag{A.23}$$

and

$$\Omega^{MP}\Omega_{NP} = \delta_N^M \,, \tag{A.24}$$

while the $\mathrm{E}_{7(7)}$ adjoint indices α, β are raised and lowered with the Killing metric

$$\kappa_{\alpha\beta} = (t_\alpha)^M{}_N \, (t_\beta)^N{}_M \,. \tag{A.25}$$

For the section condition we find [68, 76]

$$(t_\alpha)^{MN}\, \partial_M \otimes \partial_N = (t_\alpha)^{MN}\, \partial_M \partial_N = 0 \,, \qquad \Omega^{MN} \partial_M \otimes \partial_N = 0 \,. \tag{A.26}$$

Under the decomposition $\mathrm{E}_{7(7)} \longrightarrow \mathrm{SL}(8)$, we have

$$\mathbf{56} \longrightarrow \mathbf{28} \oplus \overline{\mathbf{28}} \,, \tag{A.27}$$

and therefore a generalized vector V^M can be written as

$$V^M = \left(V^{IJ},\, V_{IJ} \right) \,, \tag{A.28}$$

where $I, J = 1, \ldots, 8$ labels the fundamental of $\mathrm{SL}(8)$ and $V^{IJ} = V^{[IJ]}$ and $V_{IJ} = V_{[IJ]}$ transform in the $\mathbf{28}$ and $\overline{\mathbf{28}}$ of $\mathrm{SL}(8)$. We can further decompose $\mathrm{SL}(8) \longrightarrow \mathrm{SL}(7) \times \mathbb{R}^+$ so that

$$\begin{aligned} V^{IJ} &= \left(V^i,\, V^{ij} = \frac{1}{5!}\epsilon^{ijklmnp} V_{klmnp} \right) \,, \\ V_{IJ} &= \left(V_i,\, V_{ij} \right) \,, \end{aligned} \tag{A.29}$$

and the coordinate derivatives become

$$\partial_M = \left(\partial_i,\, \partial^{ij},\, -\frac{1}{2}\partial_{ij},\, \partial^i \right) \,. \tag{A.30}$$

Similarly, the adjoint representation of $\mathrm{E}_{7(7)}$ decomposes under $\mathrm{SL}(8)$ as

$$\mathbf{133} \longrightarrow \mathbf{63} \oplus \mathbf{70} \,. \tag{A.31}$$

In the $\mathrm{SL}(8)$ basis, the only non-zero components of the generators $(t_\alpha)_M{}^N$ are

$$\begin{aligned} \left(t_I{}^J \right)_{KL}{}^{MN} &= -\frac{1}{\sqrt{3}} \delta^J_{[K} \delta^{MN}_{L]I} - \frac{1}{8\sqrt{3}} \delta^J_I \delta^{MN}_{KL} = -\left(t^J_I \right)^{MN}{}_{KL} \,, \\ \left(t_{IJKL} \right)_{MNPQ} &= \frac{1}{24\sqrt{2}} \epsilon_{IJKLMNPQ} \,, , \qquad \left(t_{IJKL} \right)^{MNPQ} = \frac{1}{\sqrt{2}} \delta^{MNPQ}_{IJKL} \,. \end{aligned} \tag{A.32}$$

The only non-zero components of the symplectic invariant Ω_{MN} are given by

$$\Omega^{IJ}{}_{KL} = \delta^{IJ}_{KL} = -\Omega_{KL}{}^{IJ} . \tag{A.33}$$

As a result, we find for the section condition the result (8.70). For the generalized Lie derivative (A.21) corresponding to a spacetime diffeomorphism $V^M = V^{i8} = (\xi^i, 0, 0, 0)$, we find for the generalized vielbein

$$\begin{aligned}
\mathcal{L}_\xi E_{\bar{M}}{}^i &= \xi^j \partial_j E_{\bar{M}}{}^i - E_{\bar{M}}{}^j \partial_j \xi^i - 3 E_{\bar{M}\,jk} \partial^{[jk} \xi^{i]} - \frac{1}{2} E_{\bar{M}}{}^i \partial_k \xi^k - E_{\bar{M}}{}^{ij} \partial_{jk} \xi^k , \\
\mathcal{L}_\xi E_{\bar{M}\,ij} &= \xi^k \partial_k E_{\bar{M}\,ij} + 2 E_{\bar{M}\,k[j} \partial_{i]} \xi^k - \frac{1}{2} E_{\bar{M}\,[j} \partial_{i]k} \xi^k - \frac{1}{4} \epsilon_{ijklmnp} E_{\bar{M}}{}^{kl} \partial^{mn} \xi^p \\
&\quad - \frac{1}{2} E_{\bar{M}\,ij} \partial_k \xi^k , \\
\mathcal{L}_\xi E_{\bar{M}}{}^{ij} &= \xi^k \partial_k E_{\bar{M}}{}^{ij} - 2 E_{\bar{M}}{}^{k[j} \partial_k \xi^{i]} - \frac{3}{2} E_{\bar{M}\,k} \partial^{[ij} \xi^{k]} + \frac{1}{2} E_{\bar{M}}{}^{ij} \partial_k \xi^k , \\
\mathcal{L}_\xi E_{\bar{M}\,i} &= \xi^j \partial_j E_{\bar{M}\,i} + E_{\bar{M}\,j} \partial_i \xi^j + \frac{1}{2} E_{\bar{M}\,i} \partial_j \xi^j .
\end{aligned} \tag{A.34}$$

A.2 U-Dualities

We gave the derivation of the formulae for the U-dualities in section 7.2.3. Here, we state those formulae. Note that all transformations can be read in both direction – depending on what kind of fields are present.

A.2.1 U-duality along three directions

In this thesis we use parallelisable backgrounds such as the $\mathcal{N}_3^k$ and $\mathcal{N}_2^k$ spaces in sections 8.3 – 8.3 with metric

$$ds^2 = \left(e^{\bar{1}}\right)^2 + \left(e^{\bar{2}}\right)^2 + \ldots + \left(e^{\bar{7}}\right)^2 , \tag{A.35}$$

where $e^{\bar{i}} = e^{\bar{i}}{}_i dx^i$ are globally well-defined 1-forms.

In those spaces, we always have a triangular vielbein

$$e^{\bar{i}}{}_j = \delta^{\bar{i}}{}_j + N^{\bar{i}}{}_j , \qquad e^i{}_{\bar{j}} = \delta^i{}_{\bar{j}} - N^i{}_{\bar{j}} . \tag{A.36}$$

As a result, we can describe U-dualities in the three directions i, j and k by the totally anti-symmetric combination $\omega^{ijk} = \omega_{ijk} = \pm 1$. As a result we get the background

$$N^{\bar{i}}{}_j \longrightarrow \left\{ \begin{array}{ll} \Omega'^{ijk} & = 3 N^{[i}{}_{\bar{l}} \omega^{jk]l} , \\ C'_{ijk} & = -3 N^{\bar{l}}{}_{[i} \omega_{jk]\bar{l}} , \\ N'^{\bar{i}}{}_j & = N^{\bar{i}}{}_j + \tfrac{3}{2} N^{[i}{}_{\bar{m}} \omega^{kl]\bar{m}} \omega_{jkl} - \tfrac{3}{2} N^{\bar{m}}{}_{[j} \omega_{kl]\bar{m}} \omega^{\bar{i}kl} . \end{array} \right. \tag{A.37}$$

The new coordinates are given by

$$\begin{aligned} x'^i &= \frac{1}{2}\omega^{ijk}\tilde{x}_{jk} \, , \\ \tilde{x}'_{ij} &= \omega_{ijk}x^k + \frac{1}{3!}\omega^{klm}\tilde{x}_{ijklm} \, , \\ \tilde{x}'_{ijklm} &= \omega_{[ijk}\tilde{x}_{lm]} + \frac{1}{2}\omega^{npq}\epsilon_{ijklmnp}\tilde{x}_q \, , \\ \tilde{x}'_i &= \frac{1}{5!2}\omega_{ijk}\epsilon^{jklmnpq}\tilde{x}_{lmnpq} \, . \end{aligned} \tag{A.38}$$

If the non-geometric and the geometric 3-form in (A.37) vanish, we will not get any change of twisted torus by the U-duality, up to the transformations of the coordinates (A.38).

We can act with another U-duality on a space which yields a non-geometric three-form (with flat metric). As a result we find

$$\Omega^{ijk} \longrightarrow \begin{cases} \Omega'^{ijklmn} = 20\Omega^{[ijk}\,\omega^{lmn]} \, , \\ C'_{ijk} = \left(\frac{1}{3!}\omega_{lmn}\Omega^{lmn}\right)\omega_{ijk} \, . \end{cases} \tag{A.39}$$

In the case, that both of these vanish, the space is either left unchanged, up to the coordinate transformation (A.38), becomes the original twisted torus by the formulae (A.37).

Additionally, we have to exchange the momenta and wrapping modes according to the rules

$$\begin{aligned} p'_i &= \frac{1}{2}\omega_{ijk}w^{jk} \, , \\ w'^{ij} &= \omega^{ijk}p_k + \frac{1}{3!}\omega_{klm}w^{ijklm} \, , \\ w'^{ijklm} &= \omega^{[ijk}w^{lm]} + \frac{1}{2}\omega_{npq}\epsilon^{ijklmnp}w^q \, , \\ w'^i &= \frac{1}{5!2}\omega^{ijk}\epsilon_{ijklmnpq}w^{lmnpq} \, , \end{aligned} \tag{A.40}$$

which mirror the behavior of the positions in (A.38).

A.2.2 U-duality along six directions

Similar rules apply for U-dualities in six dimensions. Starting from a twisted torus with the triangular vielbein

$$e^{\bar{i}}{}_j = \delta^{\bar{i}}{}_j + N^{\bar{i}}{}_j \, , \qquad e^i{}_{\bar{j}} = \delta^i{}_{\bar{j}} - N^i{}_{\bar{j}} \, , \tag{A.41}$$

and dualizing the background along $\omega^{ijklmn} = \omega_{ijklmn} = \pm 1$, we find the background

$$N^{\bar{i}}{}_j \longrightarrow \begin{cases} \Omega'^{ijklmn} &= 6N^{[i}{}_{\bar{p}}\,\omega^{jklmn]\bar{p}} \, , \\ C'_{ijklmn} &= -6N^{\bar{p}}{}_{[i}\omega_{jklmn]\bar{p}} \, , \\ N'^{\bar{i}}{}_j &= N^{\bar{i}}{}_j + \frac{6}{5!}N^{[i}{}_{\bar{q}}\omega^{klmnp]\bar{q}}\omega_{jklmnp} - \frac{6}{5!}N^{\bar{q}}{}_{[j}\omega_{klmnp]\bar{q}}\omega^{\bar{i}klmnp} \, . \end{cases} \tag{A.42}$$

with new coordinates

$$\begin{aligned} x'^i &= \frac{1}{5!}\,\omega^{ijklmn}\tilde{x}_{jklmn}\,, \\ \tilde{x}'_{ij} &= \frac{1}{5!}\,\omega^{klmnpq}\epsilon_{ijklmnp}\tilde{x}_q\,, \\ \tilde{x}'_{ijklm} &= \omega_{ijklmn}x^n\,, \\ \tilde{x}'_i &= \frac{1}{5!}\,\omega_{ijklmn}\epsilon^{jklmnpq}\tilde{x}_{pq}\,. \end{aligned} \tag{A.43}$$

The momenta and wrapping modes mirror the behavior of the positions

$$\begin{aligned} p'_i &= \frac{1}{5!}\,\omega_{ijklmn}w^{jklmn}\,, \\ w'^{ij} &= \frac{1}{5!}\,\omega_{klmnpq}\epsilon^{ijklmnp}w^q\,, \\ w'^{ijklm} &= \omega^{ijklmn}p_n\,, \\ w'^i &= \frac{1}{5!}\,\omega^{ijklmn}\epsilon_{jklmnpq}w^{pq}\,. \end{aligned} \tag{A.44}$$

If the non-geometric and the geometric six-form in (A.42) vanish, we will not get any change of twisted torus by the U-duality, up to the transformations of the coordinates (A.44).

We can act with another U-duality on a space which yields a already a non-geometric three-form (with flat metric). As a result we find

$$C'_{ijk} = \frac{1}{3!}\,\omega_{ijklmn}\Omega^{lmn}\,. \tag{A.45}$$

In the case, that the geometric three-form vanishes, the space is left unchanged, up to the coordinate transformation (A.38).

Appendix B

Open-string computations

In this part of the appendix, we compute the results of the world-sheet computation used in chapter 9. We are going to summarize some technical details of the computation for the Seiberg-Witten mapping (9.9), as well perform the computations of two-point functions and commutators for Neumann-Neumann (NN) and Dirichlet-Dirichlet (DD) open strings.

B.1 Two-Point Function I

In the beginning, we review the computations of and Witten [120] for the two-point function of two NN open-string coordinates on the boundary. Note that the open-string coordinates $\mathsf{X}^i_{\mathrm{NN}}(z,\bar z)$ are functions on the upper half-plane. We are going to parameterize them by $z \in \mathbb{C}$ with $\mathrm{Im}\, z \geq 0$, then the boundary of the open-string world-sheet is given by $\mathrm{Im}\, z = 0$. As a starting point they use the open-string two-point function [112, 113, 114]

$$\begin{aligned}\Big\langle \mathsf{X}^i_{\mathrm{NN}}(z,\bar z)\mathsf{X}^j_{\mathrm{NN}}(z',\bar z')\Big\rangle =& -\alpha'\bigg[g^{ij}\log|z-z'| - g^{ij}\log|z-\bar z'| \\ &+ G^{ij}\log|z-\bar z'|^2 + \theta^{ij}\log\frac{z-\bar z'}{\bar z - z'} + D^{ij}\bigg] \\ =& -\alpha'\bigg[\frac{1}{2}g^{ij}\log\frac{1+y^2}{1+x^2} + G^{ij}\log\Big(1+x^2\Big) \\ &+ G^{ij}\log\left(\tau-\tau'\right)^2 + \theta^{ij}\log\frac{1+ix}{1-ix} + D^{ij}\bigg],\end{aligned} \tag{B.1}$$

where the matrices g^{ij}, G^{ij}, θ^{ij} and D^{ij} are introduced already in the mapping (9.9). We define our complex coordinates as $z = \tau + i\sigma$ and $z' = \tau' + i\sigma'$ on the world-sheet and introduce as useful combinations

$$x = \frac{\sigma+\sigma'}{\tau-\tau'}\,, \qquad y = \frac{\sigma-\sigma'}{\tau-\tau'}\,. \tag{B.2}$$

To compute the short-distance behavior of the two-point function (B.1), we have to evaluate the equations at the boundaries $\sigma = \sigma' = 0$:

$$\begin{array}{lcl} \sigma = \sigma' \to 0 & \text{such that} & |x| = \text{fixed} \gg 1\,, \\ \tau - \tau' \to 0 & & y\ = 0\,. \end{array} \tag{B.3}$$

Using a convenient choice of D^{ij}, the constant x-dependent terms in (B.1) proportional to g^{ij} and G^{ij} are canceled, whereas for large x, the θ^{ij}-term become a step function depending on the sign of x. Using

$$\epsilon(\tau - \tau') = \begin{cases} +1 & \tau > \tau' \\ -1 & \tau < \tau' \end{cases} \tag{B.4}$$

we get for the two-point function (B.1) evaluated on the boundary $\partial\Sigma$ of the world-sheet the result

$$\left\langle \mathsf{X}^i_{\text{NN}}(z,\bar{z})\mathsf{X}^j_{\text{NN}}(z',\bar{z}') \right\rangle_{\partial\Sigma} = -\alpha' G^{ij} \log{(\tau - \tau')}^2 + i\alpha'\pi\theta^{ij}\epsilon(\tau - \tau')\,. \tag{B.5}$$

As next step, we can now compute the equal-time commutator for a free theory. Using the two-point function on the boundary (B.5), the commutator of two open-string coordinates on the boundary is [120]

$$\begin{aligned} \left[\mathsf{X}^i_{\text{NN}}(\tau), \mathsf{X}^j_{\text{NN}}(\tau)\right] &= T\left(\mathsf{X}^i_{\text{NN}}(\tau)\mathsf{X}^j_{\text{NN}}(\tau^-) - \mathsf{X}^i_{\text{NN}}(\tau)\mathsf{X}^j_{\text{NN}}(\tau^+)\right) \\ &= 2\pi\alpha' i\theta^{ij}\,, \end{aligned} \tag{B.6}$$

where $T(\ldots)$ denotes time ordering.

B.2 Two-Point Function II

Using this method, we can compute the two-point function of open-string coordinates with NN and DD boundary conditions. Let us start with the mode expansion for the open-string coordinates:

$$\begin{array}{lll} \text{NN} & \mathsf{X}_{\text{NN}}(z,\bar{z}) = q - i\alpha' p \log|z|^2 & + i\sqrt{\dfrac{\alpha'}{2}} \displaystyle\sum_{n\neq 0} \frac{1}{n}\alpha_n (z^{-n} + \bar{z}^{-n})\,, \\ \text{DD} & \mathsf{X}_{\text{DD}}(z,\bar{z}) = q_0 + \dfrac{q_1 - q_0}{2\pi i} \log\dfrac{z}{\bar{z}} & + i\sqrt{\dfrac{\alpha'}{2}} \displaystyle\sum_{n\neq 0} \frac{1}{n}\alpha_n (z^{-n} - \bar{z}^{-n})\,, \end{array} \tag{B.7}$$

where q and p are the center-of-mass position and momentum for the NN string, and q_0 and q_1 denote the position of the string-endpoints of the DD string. With the algebra $[\alpha_m, \alpha_n] = m\delta_{m+n}$, we get for the two-point function

$$\left\langle \mathsf{X}_{\text{NN}}(z,\bar{z})\mathsf{X}_{\text{DD}}(z',\bar{z}') \right\rangle = -\frac{\alpha'}{2}\left(\log\frac{z - z'}{\bar{z} - \bar{z}'} + \log\frac{\bar{z} - z}{z - \bar{z}'}\right)\,. \tag{B.8}$$

Following the computations in section B.1, we are going to evaluate the two-point function on the boundary by using $\operatorname{Im} z = 0$:

$$\Big\langle \mathsf{X}_{\mathrm{NN}}(z,\bar z)\mathsf{X}_{\mathrm{DD}}(z',\bar z')\Big\rangle_{\partial\Sigma} = -\frac{\alpha'}{2}\left(\log\frac{1+iy}{1-iy} + \log\frac{1-ix}{1+ix}\right). \tag{B.9}$$

The result for the two-point function (B.8) on the boundary under the limit (B.3) is

$$\Big\langle \mathsf{X}_{\mathrm{NN}}(\tau)\mathsf{X}_{\mathrm{DD}}(\tau')\Big\rangle_{\partial\Sigma} = \frac{\pi\alpha' i}{2}\,\epsilon(\tau-\tau')\,. \tag{B.10}$$

Then, for the free theory the equal-time commutator is

$$[\mathsf{X}_{\mathrm{NN}}(\tau),\mathsf{X}_{\mathrm{DD}}(\tau)] = T\left(\mathsf{X}_{\mathrm{NN}}(\tau)\mathsf{X}_{\mathrm{DD}}(\tau^-) - \mathsf{X}_{\mathrm{NN}}(\tau)\mathsf{X}_{\mathrm{DD}}(\tau^+)\right) = \pi\alpha' i\,. \tag{B.11}$$

As a result, we find a non-vanishing commutator for the coordinates X_{NN} and X_{DD}.

B.3 Commutator for the String Coordinates

Using the mode expansions for the NN open string (B.7) and for the dual closed string

$$\tilde X_{\mathrm{closed}}(z,\bar z) = q_0 + \frac{p_0}{2\pi i}\log\frac{z}{\bar z} + i\sqrt{\frac{\alpha'}{2}}\sum_{n\neq 0}\frac{1}{n}(\alpha_n z^{-n} - \bar\alpha_n \bar z^{-n})\,, \tag{B.12}$$

we can now compute the commutator between these two. Evaluating the algebra $[\alpha_m,\alpha_n] = m\delta_{m+n}$, we find for the commutator

$$\Big[\mathsf{X}_{\mathrm{NN}}(\tau,\sigma),\tilde X_{\mathrm{closed}}(\tau',\sigma')\Big] = \frac{\alpha'}{2}\sum_{n\neq 0}\frac{1}{n}\left(z^{-n}+\bar z^{-n}\right)\left(z'^n - \bar z'^n\right)\,. \tag{B.13}$$

A careful analysis of the correct limes, leads to result

$$\Big[\mathsf{X}_{\mathrm{NN}}(\tau,0),\tilde X_{\mathrm{closed}}(\tau',0)\Big] = \begin{cases} +i\pi\alpha' & \text{if } \tau>0\,,\\ -i\pi\alpha' & \text{if } \tau<0\,.\end{cases} \tag{B.14}$$

Bibliography

[1] A. M. Turing, *Intelligent machinery, a heretical theory*, *The Turing test: Verbal behavior as the hallmark of intelligence* **105** (1948) .

[2] W. S. McCulloch and W. Pitts, *A logical calculus of the ideas immanent in nervous activity*, *The bulletin of mathematical biophysics* **5** (1943) 115.

[3] F. Rosenblatt, *The perceptron: A probabilistic model for information storage and organization in the brain*, *Psychological Review* (1958) 65.

[4] S. Linnainmaa, *Taylor expansion of the accumulated rounding error*, *BIT Numerical Mathematics* **16** (1976) 146.

[5] D. E. Rumelhart, G. E. Hinton and R. J. Williams, *Learning representations by back-propagating errors*, *nature* **323** (1986) 533.

[6] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard et al., *Backpropagation applied to handwritten zip code recognition*, *Neural computation* **1** (1989) 541.

[7] J. Bennett, S. Lanning et al., *The netflix prize*, in *Proceedings of KDD cup and workshop*, vol. 2007, p. 35, New York, NY, USA., 2007.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, *Imagenet: A large-scale hierarchical image database*, in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[9] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche et al., *Mastering the game of go with deep neural networks and tree search*, *nature* **529** (2016) 484.

[10] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby et al., *Machine learning and the physical sciences*, *Reviews of Modern Physics* **91** (2019) .

[11] P. A. M. Dirac, *Quantised Singularities in the Electromagnetic Field*, *Proceedings of the Royal Society of London Series A* **133** (1931) 60.

[12] A. Einstein, *Zur elektrodynamik bewegter körper*, *Annalen der physik* **4** (1905) .

[13] L. Eötvös, *Über die anziehung der erde auf verschiedene substanzen*, *Mathematische und naturwissenschaftliche Berichte aus Ungarn* **8** (1890) 65.

[14] A. Einstein, *Erklärung der Perihelbewegung des Merkur aus der allgemeinen Relativitätstheorie*, *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften (Berlin* (1915) 831.

[15] M. Planck, *On an improvement of wien's equation for the spectrum*, *Ann. Physik* **1** (1900) 719.

[16] A. Einstein, *Über einen die Erzeugung und Verwandlung des Lichtes betreffenden heuristischen Gesichtspunkt*, *Annalen der Physik* **322** (1905) 132.

[17] G. Aad, B. Abbott, J. Abdallah, O. Abdinov, R. Aben, M. Abolins et al., *Combined measurement of the higgs boson mass inppcollisions ats=7and 8 tev with the atlas and cms experiments*, *Physical Review Letters* **114** (2015) .

[18] W. De Boer, *Grand unified theories and supersymmetry in particle physics and cosmology*, *Progress in Particle and Nuclear Physics* **33** (1994) 201–301.

[19] R. Blumenhagen, D. Lüst and S. Theisen, *Basic concepts of string theory*, Theoretical and Mathematical Physics. Springer, Heidelberg, Germany, 2013, 10.1007/978-3-642-29497-6.

[20] F. Gliozzi, J. Scherk and D. I. Olive, *Supersymmetry, Supergravity Theories and the Dual Spinor Model*, *Nucl. Phys. B* **122** (1977) 253.

[21] P. du Bosque, *Advancements in Double & Exceptional Field Theory on Group Manifolds*, Ph.D. thesis, Munich U., 2017. `1710.01235`. 10.5282/edoc.21176.

[22] S. Krippendorf and M. Syvaeri, *Detecting Symmetries with Neural Networks*, `2003.13679`.

[23] M. Syvaeri and S. Krippendorf, *Improving simulations with symmetry control neural networks*, 2021.

[24] S. Krippendorf, D. Lüst and M. Syvaeri, *Integrability ex machina*, `2103.07475`.

[25] D. Lüst, E. Malek and M. Syväri, *Locally non-geometric fluxes and missing momenta in M-theory*, *JHEP* **01** (2018) 050 [`1710.05919`].

[26] D. Lüst, E. Malek, E. Plauschinn and M. Syväri, *Open-String Non-Associativity in an R-flux Background*, *JHEP* **05** (2020) 157 [`1903.05581`].

[27] S. Krippendorf, R. Kroepsch and M. Syvaeri, *Revealing systematics in phenomenologically viable flux vacua with reinforcement learning*, `2107.04039`.

[28] D. J. C. MacKay, *Information Theory, Inference & Learning Algorithms.* Cambridge University Press, USA, 2002.

[29] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning.* The MIT Press, 2016.

[30] P. Mehta, M. Bukov, C.-H. Wang, A. G. Day, C. Richardson, C. K. Fisher et al., *A high-bias, low-variance introduction to machine learning for physicists*, *Physics Reports* **810** (2019) 1–124.

[31] F. Ruehle, *Data science applications to string theory*, *Phys. Rept.* **839** (2020) 1.

[32] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction.* The MIT Press, second ed., 2018.

[33] S. Krippendorf, R. Kroepsch and M. Syvaeri, *Revealing systematics in phenomenologically viable flux vacua with reinforcement learning*, 2021.

[34] Y. LeCun, C. Cortes and C. Burges, *Mnist handwritten digit database*, *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* **2** (2010) .

[35] K. Hornik, M. Stinchcombe and H. White, *Multilayer feedforward networks are universal approximators*, *Neural Netw.* **2** (1989) 359–366.

[36] Z. Lu, H. Pu, F. Wang, Z. Hu and L. Wang, *The expressive power of neural networks: A view from the width*, in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 6232–6240, 2017.

[37] J. Duchi, E. Hazan and Y. Singer, *Adaptive subgradient methods for online learning and stochastic optimization*, *J. Mach. Learn. Res.* **12** (2011) 2121–2159.

[38] M. D. Zeiler, *Adadelta: An adaptive learning rate method*, *CoRR* **abs/1212.5701** (2012) .

[39] T. Tieleman and G. Hinton, "Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude." COURSERA: Neural Networks for Machine Learning, 2012.

[40] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, eds., 2015, http://arxiv.org/abs/1412.6980.

[41] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard et al., *Backpropagation applied to handwritten zip code recognition*, *Neural Comput.* **1** (1989) 541–551.

[42] H. Yakura, S. Shinozaki, R. Nishimura, Y. Oyama and J. Sakuma, *Malware analysis of imaged binary samples by convolutional neural network with attention mechanism*, pp. 127–134, 03, 2018, DOI.

[43] T. Mikolov, K. Chen, G. Corrado and J. Dean, *Efficient estimation of word representations in vector space*, *arXiv preprint arXiv:1301.3781* (2013) .

[44] Q. Zhou, P. Tang, S. Liu, J. Pan, Q. Yan and S.-C. Zhang, *Learning atoms for materials discovery*, *Proceedings of the National Academy of Sciences* **115** (2018) E6411 [`1807.05617`].

[45] L. v. d. Maaten and G. Hinton, *Visualizing data using t-sne*, *Journal of machine learning research* **9** (2008) 2579.

[46] P. Candelas, A. M. Dale, C. A. Lutken and R. Schimmrigk, *Complete Intersection Calabi-Yau Manifolds*, *Nucl. Phys.* **B298** (1988) 493.

[47] J. Gray, A. S. Haupt and A. Lukas, *All Complete Intersection Calabi-Yau Four-Folds*, *JHEP* **07** (2013) 070 [`1303.1832`].

[48] T. Hubsch, *Calabi-Yau manifolds: A Bestiary for physicists.* World Scientific, Singapore, 1994.

[49] P. Green and T. Hubsch, *Calabi-yau Manifolds as Complete Intersections in Products of Complex Projective Spaces*, *Commun. Math. Phys.* **109** (1987) 99.

[50] S. Kearnes, K. McCloskey, M. Berndl, V. Pande and P. Riley, *Molecular graph convolutions: moving beyond fingerprints*, *Journal of Computer-Aided Molecular Design* **30** (2016) 595 [`1603.00856`].

[51] P. S. Green, T. Hubsch and C. A. Lutken, *All Hodge Numbers of All Complete Intersection Calabi-Yau Manifolds*, *Class. Quant. Grav.* **6** (1989) 105.

[52] S. Greydanus, M. Dzamba and J. Yosinski, *Hamiltonian neural networks*, *CoRR* **abs/1906.01563** (2019) [`1906.01563`].

[53] P. Battaglia, R. Pascanu, M. Lai, D. J. Rezende et al., *Interaction networks for learning about objects, relations and physics*, in *Advances in neural information processing systems*, pp. 4502–4510, 2016.

[54] M. Cranmer, R. Xu, P. Battaglia and S. Ho, *Learning symbolic physics with graph networks*, *ArXiv* **abs/1909.05862** (2019) .

[55] D. J. Rezende, S. Racanière, I. Higgins and P. Toth, *Equivariant hamiltonian flows*, *arXiv preprint arXiv:1909.13739* (2019) .

[56] M. Syvaeri and S. Krippendorf, *Symmetry control neural networks*, 2021.

[57] N. Beisert, *Integrability in QFT and AdS/CFT*, `https://edu.itp.phys.ethz.ch/hs13/13HSInt/IntHS13Notes.pdf`.

[58] E. Abdalla, M. C. B. Abdalla and K. D. Rothe, *Non-Perturbative Methods in 2 Dimensional Quantum Field Theory.* WORLD SCIENTIFIC, 2nd ed., 2001, 10.1142/4678, [`https://www.worldscientific.com/doi/pdf/10.1142/4678`].

[59] K. Pohlmeyer, *Integrable Hamiltonian Systems and Interactions Through Quadratic Constraints*, *Commun. Math. Phys.* **46** (1976) 207.

[60] A. Torrielli, *Lectures on Classical Integrability*, *J. Phys. A* **49** (2016) 323001 [`1606.02946`].

[61] D. S. Berman and C. D. A. Blair, *The Geometry, Branes and Applications of Exceptional Field Theory*, *Int. J. Mod. Phys. A* **35** (2020) 2030014 [`2006.09777`].

[62] G. Aldazabal, D. Marqués and C. Núñez, *Double Field Theory: A Pedagogical Review*, *Class.Quant.Grav.* **30** (2013) 163001 [`1305.1907`].

[63] T. Buscher, *A Symmetry of the String Background Field Equations*, *Phys.Lett.* **B194** (1987) 59.

[64] T. Buscher, *Path Integral Derivation of Quantum Duality in Nonlinear Sigma Models*, *Phys.Lett.* **B201** (1988) 466.

[65] E. Cremmer, B. Julia and J. Scherk, *Supergravity in theory in 11 dimensions*, *Physics Letters B* **76** (1978) 409.

[66] E. Cremmer and B. Julia, *The* $n = 8$ *supergravity theory. i. the lagrangian*, *Physics Letters B* **80** (1978) 48.

[67] E. Cremmer and B. Julia, *The so(8) supergravity*, *Nuclear Physics B* **159** (1979) 141.

[68] D. S. Berman, M. Cederwall, A. Kleinschmidt and D. C. Thompson, *The gauge structure of generalised diffeomorphisms*, *JHEP* **01** (2013) 064 [`1208.5884`].

[69] D. S. Berman and M. J. Perry, *Generalized Geometry and M theory*, *JHEP* **06** (2011) 074 [`1008.1763`].

[70] O. Hohm and H. Samtleben, *Exceptional Form of D=11 Supergravity*, *Phys.Rev.Lett.* **111** (2013) 231601 [`1308.1673`].

[71] D. S. Berman, C. D. A. Blair, E. Malek and F. J. Rudolph, *An action for F-theory:* $\mathrm{SL}(2) \times \mathbb{R}^+$ *exceptional field theory*, *Class. Quant. Grav.* **33** (2016) 195009 [`1512.06115`].

[72] O. Hohm and Y.-N. Wang, *Tensor hierarchy and generalized Cartan calculus in $SL(3) \times SL(2)$ exceptional field theory*, *JHEP* **04** (2015) 050 [`1501.01600`].

[73] E. T. Musaev, *Exceptional field theory: $SL(5)$*, *JHEP* **02** (2016) 012 [`1512.02163`].

[74] A. Abzalov, I. Bakhmatov and E. T. Musaev, *Exceptional field theory: $SO(5,5)$*, *JHEP* **06** (2015) 088 [`1504.01523`].

[75] O. Hohm and H. Samtleben, *Exceptional Field Theory I: $E_{6(6)}$ covariant Form of M-Theory and Type IIB*, *Phys. Rev.* **D89** (2014) 066016 [`1312.0614`].

[76] O. Hohm and H. Samtleben, *Exceptional Field Theory II: $E_{7(7)}$*, *Phys.Rev.* **D89** (2014) 066017 [`1312.4542`].

[77] O. Hohm and H. Samtleben, *Exceptional Field Theory III: $E_{8(8)}$*, *Phys.Rev.* **D90** (2014) 066002 [`1406.3348`].

[78] E. Malek, *U-duality in three and four dimensions*, *Int. J. Mod. Phys.* **A32** (2017) 1750169 [`1205.6403`].

[79] E. Plauschinn, *Non-geometric backgrounds in string theory*, `1811.11203`.

[80] J. Shelton, W. Taylor and B. Wecht, *Nongeometric flux compactifications*, *JHEP* **10** (2005) 085 [`hep-th/0508133`].

[81] M. Grana, R. Minasian, M. Petrini and D. Waldram, *T-duality, Generalized Geometry and Non-Geometric Backgrounds*, *JHEP* **04** (2009) 075 [`0807.4527`].

[82] D. Andriot, M. Larfors, D. Lüst and P. Patalong, *A ten-dimensional action for non-geometric fluxes*, *JHEP* **09** (2011) 134 [`1106.4015`].

[83] D. Andriot, O. Hohm, M. Larfors, D. Lüst and P. Patalong, *A geometric action for non-geometric fluxes*, *Phys.Rev.Lett.* **108** (2012) 261602 [`1202.3060`].

[84] D. Andriot, O. Hohm, M. Larfors, D. Lüst and P. Patalong, *Non-Geometric Fluxes in Supergravity and Double Field Theory*, *Fortsch.Phys.* **60** (2012) 1150 [`1204.1979`].

[85] R. Blumenhagen, A. Deser, E. Plauschinn, F. Rennecke and C. Schmid, *The Intriguing Structure of Non-geometric Frames in String Theory*, `1304.2784`.

[86] D. Andriot and A. Betz, *β-supergravity: a ten-dimensional theory with non-geometric fluxes, and its geometric framework*, *JHEP* **12** (2013) 083 [`1306.4381`].

[87] A. Chatzistavrakidis, A. Deser and L. Jonke, *T-duality without isometry via extended gauge symmetries of 2D sigma models*, *JHEP* **01** (2016) 154 [`1509.01829`].

[88] C. D. A. Blair and E. Malek, *Geometry and fluxes of SL(5) exceptional field theory*, *JHEP* **03** (2015) 144 [`1412.0635`].

[89] H. Samtleben and M. Weidner, *The Maximal D=7 supergravities*, *Nucl.Phys.* **B725** (2005) 383 [`hep-th/0506237`].

[90] M. Günaydin, D. Lüst and E. Malek, *Non-associativity in non-geometric string and M-theory backgrounds, the algebra of octonions, and missing momentum modes*, *JHEP* **11** (2016) 027 [`1607.06474`].

[91] S. Kachru, M. B. Schulz, P. K. Tripathy and S. P. Trivedi, *New supersymmetric string compactifications*, *JHEP* **03** (2003) 061 [`hep-th/0211182`].

[92] J. Lee, *Introduction to topological manifolds*, vol. 202. Springer Science & Business Media, 2010.

[93] R. Blumenhagen and E. Plauschinn, *Nonassociative Gravity in String Theory?*, *J. Phys.* **A44** (2011) 015401 [`1010.1263`].

[94] D. Lüst, *T-duality and closed string non-commutative (doubled) geometry*, *JHEP* **1012** (2010) 084 [`1010.1361`].

[95] R. Blumenhagen, A. Deser, D. Lüst, E. Plauschinn and F. Rennecke, *Non-geometric Fluxes, Asymmetric Strings and Nonassociative Geometry*, *J. Phys.* **A44** (2011) 385401 [`1106.0316`].

[96] R. Blumenhagen, *Nonassociativity in String Theory*, in *Strings, gauge fields, and the geometry behind: The legacy of Maximilian Kreuzer*, A. Rebhan, L. Katzarkov, J. Knapp, R. Rashkov and E. Scheidegger, eds., pp. 213–224, (2011), `1112.4611`, DOI.

[97] D. Lüst, *Twisted Poisson Structures and Non-commutative/non-associative Closed String Geometry*, *PoS* **CORFU2011** (2011) 086 [`1205.0100`].

[98] I. Bakas and D. Lüst, *3-Cocycles, Non-Associative Star-Products and the Magnetic Paradigm of R-Flux String Vacua*, *JHEP* **01** (2014) 171 [`1309.3172`].

[99] C. Condeescu, I. Florakis and D. Lüst, *Asymmetric Orbifolds, Non-Geometric Fluxes and Non-Commutativity in Closed String Theory*, *JHEP* **04** (2012) 121 [`1202.6366`].

[100] D. Andriot, M. Larfors, D. Lüst and P. Patalong, *(Non-) commutative closed string on T-dual toroidal backgrounds*, *JHEP* **1306** (2013) 021 [`1211.6437`].

[101] C. D. A. Blair, *Non-commutativity and non-associativity of the doubled string in non-geometric backgrounds*, *JHEP* **1506** (2015) 091 [`1405.2283`].

[102] I. Bakas and D. Lüst, *T-duality, Quotients and Currents for Non-Geometric Closed Strings*, *Fortsch. Phys.* **63** (2015) 543 [`1505.04004`].

[103] D. Mylonas, P. Schupp and R. J. Szabo, *Membrane Sigma-Models and Quantization of Non-Geometric Flux Backgrounds*, *JHEP* **1209** (2012) 012 [`1207.0926`].

[104] P. Bouwknegt and V. Mathai, *D-branes, B fields and twisted K theory*, *JHEP* **03** (2000) 007 [`hep-th/0002023`].

[105] P. Bouwknegt, K. Hannabuss and V. Mathai, *Nonassociative tori and applications to T-duality*, *Commun. Math. Phys.* **264** (2006) 41 [`hep-th/0412092`].

[106] V. Mathai and J. M. Rosenberg, *T duality for torus bundles with H fluxes via noncommutative topology*, *Commun. Math. Phys.* **253** (2004) 705 [`hep-th/0401168`].

[107] V. Mathai and J. M. Rosenberg, *On Mysteriously missing T-duals, H-flux and the T-duality group*, in *Differential geometry and physics. Proceedings, 23rd International Conference, Tianjin, China, August 20-26, 2005*, pp. 350–358, 2004, `hep-th/0409073`.

[108] L. Freidel, R. G. Leigh and D. Minic, *Intrinsic non-commutativity of closed string theory*, *JHEP* **09** (2017) 060 [`1706.03305`].

[109] L. Freidel, R. G. Leigh and D. Minic, *Noncommutativity of closed string zero modes*, *Phys. Rev.* **D96** (2017) 066003 [`1707.00312`].

[110] L. Cornalba and R. Schiappa, *Nonassociative star product deformations for D-brane world volumes in curved backgrounds*, *Commun. Math. Phys.* **225** (2002) 33 [`hep-th/0101219`].

[111] M. Herbst, A. Kling and M. Kreuzer, *Star products from open strings in curved backgrounds*, *JHEP* **09** (2001) 014 [`hep-th/0106159`].

[112] E. S. Fradkin and A. A. Tseytlin, *Nonlinear Electrodynamics from Quantized Strings*, *Phys. Lett.* **163B** (1985) 123.

[113] A. Abouelsaood, C. G. Callan, Jr., C. R. Nappi and S. A. Yost, *Open Strings in Background Gauge Fields*, *Nucl. Phys.* **B280** (1987) 599.

[114] C. G. Callan, Jr., C. Lovelace, C. R. Nappi and S. A. Yost, *String Loop Corrections to beta Functions*, *Nucl. Phys.* **B288** (1987) 525.

[115] S. Demulder, F. Hassler and D. C. Thompson, *Doubled aspects of generalised dualities and integrable deformations*, *JHEP* **02** (2019) 189 [`1810.11446`].

[116] S. Demulder, F. Hassler, G. Piccinini and D. C. Thompson, *Integrable deformation of $\mathbb{CP}^n$ and generalised Kähler geometry*, *JHEP* **10** (2020) 086 [`2002.11144`].

[117] S. Kachru, R. Kallosh, A. Linde and S. P. Trivedi, *de sitter vacua in string theory*, *Physical Review D* **68** (2003) .

[118] D. Boyda, G. Kanwar, S. Racanière, D. J. Rezende, M. S. Albergo, K. Cranmer et al., *Sampling using su(n) gauge equivariant flows*, *Physical Review D* **103** (2021) .

[119] A. Coimbra, C. Strickland-Constable and D. Waldram, *Supergravity as Generalised Geometry II: $E_{d(d)} \times \mathbb{R}^+$ and M theory*, *JHEP* **1403** (2014) 019 [`1212.1586`].

[120] N. Seiberg and E. Witten, *String theory and noncommutative geometry*, *JHEP* **09** (1999) 032 [`hep-th/9908142`].