Study of the Cosmic Ray Composition Sensitivity of AugerPrime

-

Probing the Prospects of the Upgrade to the Pierre Auger Observatory with a Deep Learning Approach

Dissertation

zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

der Fakultät Mathematik und Naturwissenschaften

Fachgruppe Physik

der Bergischen Universität Wuppertal

vorgelegt von

M. Sc. Sonja Mayotte

(geboren in Dortmund)

June 2021

The PhD thesis can be quoted as follows:

urn:nbn:de:hbz:468-20210812-123827-3
[http://nbn-resolving.de/urn/resolver.pl?urn=urn%3Anbn%3Ade%3Ahbz%3A468-20210812-123827-3]

DOI: 10.25926/99h7-fc44
[https://doi.org/10.25926/99h7-fc44]

1. Gutachter:
Prof. Dr. Karl-Heinz Kampert
Bergische Universität Wuppertal

2. Gutachter:
Prof. Dr. Antonio Bueno Villar
Universidad de Granada

To my Moogie and Nagus,
for providing me with the lobes for business.

Abstract

The composition of cosmic ray primaries at the highest energies is one of the biggest open questions in astroparticle physics, and is directly related to understanding the sources of ultra-high energy cosmic rays. At the Pierre Auger Observatory, cosmic rays are studied via the extensive air showers they produce in the Earth's atmosphere. Improving the understanding of primary composition at the highest energies is the biggest motivation behind the upgrade of the Observatory, AugerPrime. The Surface Detector of the Observatory is, as part this upgrade, in the process of receiving new electronics and additional detectors. Most important is the scintillator detector, which will be placed on top of the existing water Cherenkov detectors. It will allow for a complementary measurement of air showers due to the differing responses of the scintillators and water Cherenkov detectors to the different shower components. This in turn will enhance the resolution the Surface Detector has for measuring the primary composition, perhaps even to an event-by-event level.

In this thesis, development of the new hardware while being tested in the Engineering Array over the last few years, is analyzed for the first time. The performance of these detector prototypes, especially in regards to the timing resolution, is studied. Using information gained from these studies, a machine learning approach based on detector simulations which include the AugerPrime upgrade is undertaken, with the goal of reconstructing the depth of shower maximum, as it is an indicator of the primary mass. Specifically, a convolutional neural network is developed to extract composition information from the difference in the signal pulses of the two different detector types. The improvement in composition resolution achieved by adding the trace information from the new detector is gauged by training the same network both with and without the scintillator detector traces and comparing the results. From this, it is shown that improvements to the overall mass resolution and event-by-event reconstruction accuracy can be expected after the completion of the AugerPrime upgrade.

Zusammenfassung

Die Zusammensetzung der Primärteilchen kosmischer Strahlung im Bereich der höchsten Energien ist eine der größten offenen Fragen im Gebiet der Astroteilchenphysik und hängt direkt mit dem Verständnis der Quellen der ultra-hochenergetischen kosmischen Strahlung zusammen. Am Pierre-Auger-Observatorium wird kosmische Strahlung mit Hilfe von Luftschauern untersucht, die die Teilchen in der Erdatmosphäre hervorrufen. Die Zusammensetzung der Primärteilchen ultra-hochenergetischer kosmischer Strahlung besser zu verstehen stellt die größte Motivation für die Aufrüstung AugerPrime dar. Dazu wird der Oberflächendetektor des Observatoriums mit neuer Elektronik und zusätzlichen Detektoren ausgestattet. Am wichtigsten hierbei ist ein Szintillationsdetektor, der auf den bereits existierenden Wasser-Cherenkov-Detektoren montiert wird. Durch die unterschiedlichen Signale des Szintillators im Vergleich zum Wasser-Cherenkov-Detektor, ergibt sich eine komplementäre Messung von Luftschauern. Dadurch wird sich eine Verbesserung in der Auflösungsgenauigkeit des Oberflächendetektors gegenüber der Zusammensetzung der Primärteilchen, vielleicht sogar für einzelne Ereignisse, ergeben.

In dieser Dissertation wurden die neuen Detektoren und ihre Elektronik, sowie Anpassungen dieser im Laufe der letzten Jahre, anhand von Prototyp-Stationen erstmals untersucht. Insbesondere die Genauigkeit der Zeitmessung der Detektoren stand dabei im Vordergrund. Unter Verwendung von Erkenntnissen aus dieser Analyse, wurde maschinelles Lernen anhand von Detektor Simulationen mit AugerPrime angewendet, um die atmosphärische Tiefe des Luftschauer-Maximums zu rekonstruieren. Hierbei handelt es sich um eine Größe zur Messung der Primärteilchenmasse. Ein Convolutional Neural Network wurde entwickelt, um aus den Unterschieden in den Signalen der zwei verschiedenen Detektoren Informationen über die Zusammensetzung der kosmischen Strahlung zu gewinnen. Die Verbesserung der Messgenauigkeit dieser Zusammensetzung mit Hilfe des zusätzlichen Signals durch den Szintillator wird untersucht, indem das Netzwerk einmal mit und einmal ohne dieses Signal trainiert und beide Modelle verglichen werden. Anhand dessen kann gezeigt werden, dass Verbesserungen sowohl in der Auflösungsgenauigkeit nach der Fertigstellung von AugerPrime zu erwarten sind.

Contents

List of Figures

List of Tables

List of Abbreviations

ADST Advanced Data Summary Tree

ANN Artificial neural network

CDAS Central Data Acquisition System

CMB Cosmic Microwave Background

CNN convolutional neural network

CR cosmic ray

DNN deep neural network

EA Engineering Array

EAS extensive air showers

EBL extragalactic background light

FADC flash analog-to-digital converter

FD Fluorescence Detector

GZK limit Greisen–Zatsepin–Kuzmin limit

HEAT high elevation fluorescence telescopes

HG high gain

HV high voltage

KASCADE KArlsruhe Shower Core and Array DEtector

LDF lateral distribution function

LG low gain

MC Monte Carlo

MIP minimum ionizing particles

NN neural network

PMT photo-multiplier tube

PPA pre-production array

PyIK Python Instrumentation Kit

RD Radio Detector

SD Surface Detector

SGD stochastic gradient descent

SNR supernova-remnant

SSD surface scintillator detector

UB unified board

UHECR ultra-high energy cosmic ray

UMD Underground Muon Detector

UUB upgraded unified board

VCT vertical and central through-going muon

VEM vertical-equivalent muon

WCD water Cherenkov detectors

1

Introduction: Astroparticle Physics at the Highest Energies

> *The molecules of your body are the same molecules that make up this station, and the nebula outside, that burn inside the stars themselves. We are star stuff. We are the universe made manifest, trying to figure itself out.*
>
> — Ambassador Delenn, *Babylon 5*

The universe has always been a source of fascination and awe for humans. Astroparticle physics is a branch of physics that developed out of this strong interest, and has been a major field of study for more than a hundred years. Despite this long history, there are still many open question across the field. A central one, that this thesis hopes to help to address, is "What are *ultra-high energy cosmic rays* (UHECRs), the highest energy particles ever observed, composed of?" This is a particularly important question, as only with a good grasp on UHECR composition it is possible to shed light on their acceleration mechanisms or back-track them from Earth through the magnetic fields to their sources.

The most prominent experiment for studying *cosmic rays* (CRs) at the highest energies is the Pierre Auger Observatory in Argentina. It is particularly powerful as it has both a Surface Detector which is always watching for the particles that reach the ground from UHECR interactions in the atmosphere, and a Fluorescence Detector which can directly observe those interactions as the shower develops in the atmosphere. Because it directly observes the shower evolution, the Fluorescence Detector of the Observatory currently has the highest shower-by-shower primary particle mass resolution. However, because it can only measure during very dark and clear nights, it has a rather low duty cycle which results in low statistics at the highest energies. To address this problem, the upgrade of the Observatory, AugerPrime, is being built to improve the primary mass resolution of the Surface Detector, and aims to do so by adding surface scintillators on top of the existing water Cherenkov detectors. The combination of these two detector types provides enhanced mass sensitivity, because the two detectors have differing responses to the different components in extensive air showers and the relative ratio of these components is a strong indicator of primary mass.

In this thesis, the performance of prototype upgraded stations is studied over the first two years of data taking. The expertise gained from the analyses carried out on this data is then used to develop a deep learning approach to reconstructing the depth of shower

maximum, X_{max}, which is an indicator of primary mass, based on AugerPrime simulations. Two neural networks are trained and compared, one with and one without scintillator detector information, in order to gauge the prospects of enhancing the shower-by-shower composition sensitivity of the Surface Detector with AugerPrime.

The thesis is split into eight chapters. After this introduction, chapter 2 gives an overview of cosmic ray physics at the highest energies. Briefly, to frame the importance of UHECR composition, the energy spectrum of cosmic rays is discussed with a description of its features and their origin. Special focus is given to the composition around the so-called ankle of the spectrum and beyond, as this is the energy range studied in this thesis. Extensive air showers are introduced as they are the phenomenon used by the Observatory to indirectly measure cosmic rays. This includes an overview of the different shower components as well as the main characteristics used to define an extensive air shower.

Simulated measurements of UHECRs made by the Pierre Auger Observatory serve as the basis of this research and as a result a good understanding of the different components used for the detection of UHECRs at the Observatory is needed. Because of this, chapter 3 gives an overview of the instrumentation and software used at the Observatory. Particular attention is given to Surface Detector data acquisition, calibration, and event reconstruction, which are described in detail. The upgrade AugerPrime, which is currently underway, is also introduced and all components are described, with a focus on the new scintillator detectors and the electronics upgrade, which are the core components needed to enhance the composition resolution of the Surface Detector.

A first set of 12 upgraded Surface Detector stations was deployed around the time the research of this thesis began. To provide a firm base for the later analysis of Surface Detector stations in simulation, in chapter 4 the data from these prototype stations is analyzed to profile their performance. This was a crucial step for both selecting possible parameters for the deep neural network to train on, and for understanding the performance of the trained network. As such, in-depth studies were performed to evaluate the timing accuracy of the detectors, the baseline stability of the signal pulses, and their detection efficiency with respect to the zenith angle.

To serve as a background for the later chapters, an overview of artificial neural networks as a method in general is provided in chapter 5. This includes a description of their basic components and the mechanisms with which they learn. As they serve as the backbone of the network architecture developed in this thesis, Cartesian and hexagonal convolution methods are described in detail, along with a quick overview of the other general layer types needed in this work.

Chapter 6 describes the detector simulation process and data pre-processing steps. The parameters used to train the neural network are selected. This includes the two detector traces of the upgraded Surface Detector stations, the reconstructed energy and zenith

angle, and the arrival times of the shower in each station. Since the input data for the neural network requires it to have certain array shapes, the data is slightly augmented before being fed into the neural network for training. Additionally, a method developed to improve the performance of the network, a rotation of the mapping for each event, is introduced. This serves as a method of increasing training speed by reducing the complexity of the network.

The network architecture developed for this thesis is presented in chapter 7. It is then used to train two models, one with and one without the data from the new scintillator detectors, which are then used to predict the X_{max} of simulated showers. Two additional methods of data augmentation are introduced and applied during training. One serves as a means of bootstrapping the training data by randomly simulating non-functioning Surface Detector stations, while the other method aims to help the network give more attention to under-represented X_{max} values. The two models trained from this are then evaluated based on their composition dependent reconstruction biases $\langle X_{\mathrm{max}} \rangle$ and their overall reconstruction resolution $\sigma\left(X_{\mathrm{max}}\right)$. After correcting for zenith dependent biases, the two models are compared and a definite improvement in composition sensitivity through the addition of the new detector is found. This is also quantified by investigating the merit factors of both models, which show that the model trained with both detector traces has a greater ability to distinguish between proton and iron induced air showers, across all energies.

The results found are finally summarized in chapter 8. Here, both models are compared to another deep learning approach developed within the collaboration, which uses only water Cherenkov detector information. It is also evident there that the additional scintillator information increases the composition sensitivity of the Surface Detectors further. A short outlook on the future prospects of the research presented in this thesis is given and a few further improvements to the pre-processing and the network architecture are proposed.

2

Cosmic Rays and Extensive Air Showers

> *I canna' change the laws of physics, Captain! A've got to have thirty minutes.*
>
> — Montgomery "Scotty" Scott, *Star Trek: The Original Series*

The existence of CRs was first hypothesized at the start of the 1900s, shortly after the discovery of radioactivity by Henri Beqcuerel in 1896 [1]. In 1900 and 1901 Julius Elster and Hans Friedrich Geitel [2, 3], and independently, Charles Thomson Rees Wilson [4] discovered that a residual radioactivity in the surrounding environment was the cause for the spontaneous ionization of air in isolated chambers, such as in an electroscope. As a direct result of these experiments the isolation of electroscopes was improved. In additional experiments, Elster and Geitel discovered a drop in ionization when taking their experiment below ground to the bottom of a salt mine. They concluded that, while the Earth seems to be the source of the radiation, certain soils and waters can act as a screening mechanism.

Between 1909 and 1910 Theodor Wulf performed experiments, with electroscopes he had further improved, on top of the Eiffel tower ($\approx 300\,\mathrm{m}$) and compared the results to ground level measurements. From the Elster and Geitel experiments, the believe at this time was that the radiation causing ionization was coming from the upper Earth layer. Wulf did measure a drop in ionization with increased height. However, the decrease was too small to support the hypothesis of radiation coming from the Earth, as an exponential decrease would be expected. He attributed these variations in γ-radiation to differences in air pressure and deemed effects of radiation from the atmosphere to be too small to be detectable with his methods [5, 6].

The final breakthrough and discovery of CRs came in 1912, when Victor Hess conducted seven balloon flights up to heights of 5200 m. His results with electroscopes showed that the ionization, after reaching a minimum, increased again considerably with height, from which he concluded that this increase must originate from radiation coming from above and is possibly of extra-terrestrial origin [7].

The term cosmic rays is used nowadays to describe high energy charged particles that traverse the universe at very nearly the speed of light. CRs, together with photons, neutrinos, and dark matter form the core of the modern field of research, astroparticle physics. CRs, photons, and neutrinos, combined with gravitation waves are the dominant phenomena studied in the field of multi-messenger physics. In contrast to photons and

neutrinos, which naturally point back to their source, the paths of CRs are heavily affected by both *Galactic* and *extragalactic magnetic fields* due to their charge. This results in their flux at Earth being measured as nearly isotropic at all except the very highest energies, diminishing the possibility of identifying their source. Additionally, at high energies their flux becomes so low that they cannot be directly detected in space and have to be observed from Earth. This makes determining their composition very difficult, as they generally break up in the upper atmosphere. This provides a very challenging and interesting environment for astroparticle researchers, as the composition of CRs is fundamental to the understanding of the processes with which they are accelerated. This knowledge in turn would help to better the understanding of the high energy universe, since UHECRs are the highest energy particles ever observed.

2.1 The Flux of Cosmic Rays

The energy spectrum of CRs relates their flux at Earth to their energy and is a powerful tool for understanding CR sources and acceleration mechanisms through the features it displays. The all-particle energy spectrum of CRs as observed by air shower experiments from the TeV range up to the highest energies of 3×10^{20} eV is shown in Fig. 2.1. Generally the flux can be described by a broken power law, where each part of the spectrum can be described with the form

$$\frac{dN}{dE} \propto E^{-\gamma}\,, \tag{2.1}$$

where γ is the spectral index. The changes in γ in the different energy regions of the spectrum are thought to occur due to changes in composition, sources, or propagation. To help highlight features in the spectrum, the flux is usually scaled with the energy, in the case of Fig. 2.1 a factor of $E^{2.6}$ is used.

2.1.1 Features in the Cosmic Ray Energy Spectrum

At the lowest energies, the flux of CRs is high enough to be directly observed at high altitudes or in space, allowing for precise composition measurements. Up to energies in the GeV range, the majority of detected CRs originate from the sun, as low energy galactic CRs are mostly blocked from the solar system by the termination shock which forms when the solar wind meets the *interstellar medium*. Beyond a few GeV, the sun reaches its maximum acceleration power for CRs and its contribution to the spectrum drops off. With increasing energy, direct measurements of CR composition have shown that galactic sources dominate the flux, revealing a composition which closely follows the atomic abundances found in stars. A slight tendency toward lighter elements is measured, most importantly in B, Be and Li, which are not produced in stellar nucleosynthesis.

Instead, these are almost entirely produced by spallation of the plentiful C, N, or O CRs, on atoms of interstellar media, resulting in the loss of 1-2 protons. In these regions the CR spectrum is well described by a power law of spectral index of $\gamma \sim 2.7$. With increasing energy the sources are believed to shift to *supernovae* and *supernova-remnants* (SNRs).

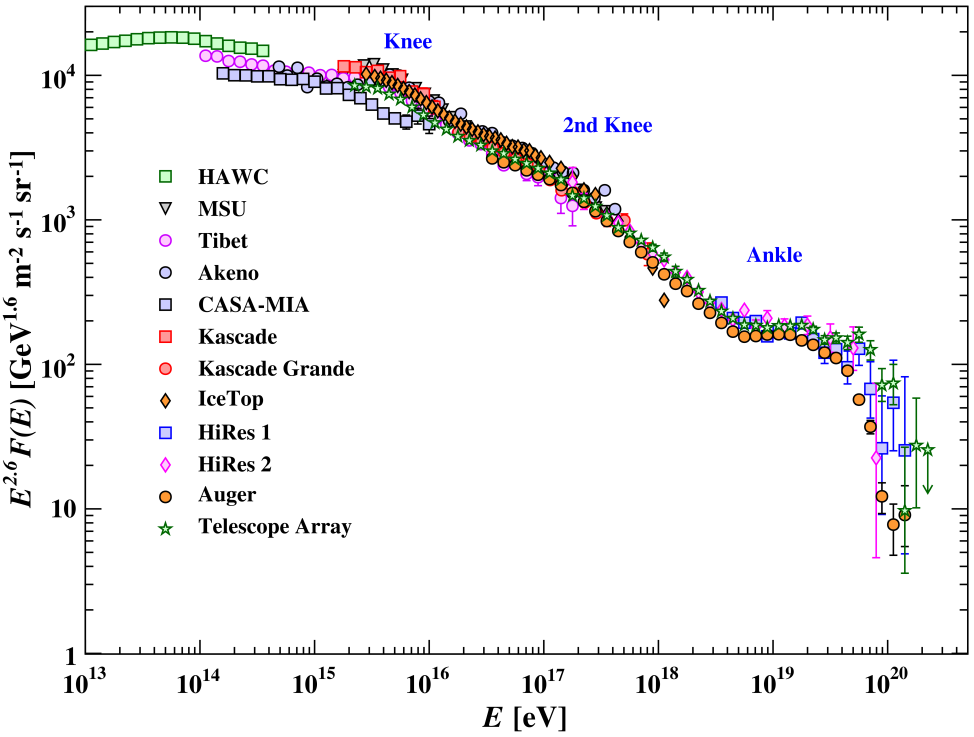


Figure 2.1: All particle energy spectrum of CRs, as measured by various different air-shower experiments over several decades of energy. Most prominent features of the spectrum are labeled at their respective energies. Taken from [8].

At around 3×10^{15} eV, a steepening of the spectrum occurs, referred to as the *knee*, which is shown in Fig. 2.1. This is believed to represent the maximum acceleration energy galactic sources can reach for protons. After that, galactic accelerators reach their maximum energies for higher and higher charges and the composition transitions from light to heavy. The behavior in this region is less stable, but can still be well described by a power law with $\gamma \sim 3.1$. A second steepening occurs at around 10^{17} eV which is closely related to these interpretations. It is referred to as the *second knee*, and is theorized to be where this maximum acceleration effect described earlier occurs for iron. There, the spectral index changes to $\gamma \sim 3.3$. Based on data from various experiments it is believed that these changes in spectral index of the spectrum occur at the same rigidity for each of the differently charged particles, and therefore occur at different energies [8]. This is around the last point where a grasp of the acceleration mechanisms and probable sources of CRs still exists, in the form of acceleration in supernovae and SNRs via first and second order Fermi acceleration [9, 10]. Additionally, around this energy cosmic rays are no longer confined to the Galaxy by the Galactic magnetic field which may also contribute to the

steepening of the spectrum. An in-depth description of galactic sources and acceleration will not be covered here, but can be found in [11].

At energies beyond 10^{17} eV, the flux of CRs enters a transition range, where possible galactic source candidates become few, and CRs from extragalactic sources likely start to contribute a significant fraction of the observed flux. As mentioned earlier, the spectral index here is well characterized by $\gamma \sim 3.3$, with this region extending up to energies of around 5×10^{18} eV. At this point, the spectrum hardens noticeably to a spectral index of $\gamma \sim 2.5$, creating what is referred to as the *ankle*. Slightly below the ankle the composition of CRs is light and has a significant proton fraction. As energy increases the composition shifts to increasingly heavier primaries [12, 13, 14] and extragalactic sources are believed to dominate the flux.

The most recent measurements from the Pierre Auger Observatory can be seen in Fig. 2.2 and clearly show this behavior of the flux at the ankle and beyond [15, 16]. The composition of the regions shown in Fig. 2.2 is of particular interest to this thesis and will be explained in more detail in section 2.3. Additional features are visible, with a newly found softening of the spectrum to $\gamma \sim 3.05$ at around 13 EeV, followed by an even stronger softening at 46 EeV to a spectral index of $\gamma \sim 5.1$, which is commonly referred to as the *cutoff*.

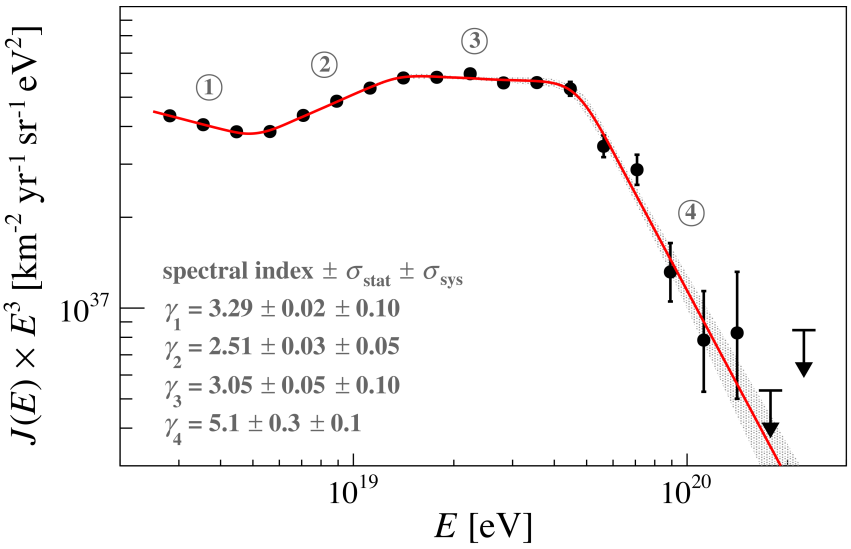


Figure 2.2: All particle energy spectrum of CRs, as measured by the Pierre Auger Observatory at the highest energies around and above the ankle. Numbered spectral indices γ_i correspond to numbered regions. Taken from [16].

2.1.2 The Cutoff of the Cosmic Ray Energy Spectrum

The phenomenological reason for the cutoff is a matter of current debate and intense research. Due to the belief that the spectrum of CRs is, at these energies, purely extragalactic in source, the origin of the cutoff is anticipated to be extragalactic as well. There

are a few possibilities currently being investigated as the probable cause:

Maximum Rigidity: This effect would be similar in nature to the one described above for the knee and second knee, only this time for extragalactic CRs, where sources accelerate particles to maximum energies according to their charge, i.e. to the same maximum rigidity.

If the flux around the ankle is proton dominated, this scenario would lead to the proton contribution in the spectrum dropping off after their maximum rigidity is reached and gradually shift to heavier and heavier components. These would similarly drop off at the same maximum rigidity, which is shifted in energy according to their charge Z. In this scenario, the ultra-high energy end of the all-particle spectrum before the cutoff would be dominated by the heaviest available component, i.e. the iron group. The suppression of the flux at the highest energies would then indicate a cutoff of the source spectrum, where extragalactic sources of UHECRs have reached this maximum rigidity to which they are able to accelerate particles even for the heaviest components [17].

GZK Limit: This effect was first theorized by Kenneth Greisen [18], Georgiy Zatsepin and Vadim Kuzmin[19], and describes interactions between high energy protons and *Cosmic Microwave Background* (CMB) photons. A proton traveling at a high Lorentz factor will see photons from the CMB approaching it as highly blue shifted and thus at high energy. A high energy photon interacting with a proton can create an unstable state known as Δ^+ resonance, which quickly decays via two channels

$$\gamma_{\text{CMB}} + p \to \Delta^+ \to p + \pi^0 \tag{2.2}$$

$$\gamma_{\text{CMB}} + p \to \Delta^+ \to n + \pi^+ \,. \tag{2.3}$$

The created pions then decay further via the known pion decay channels [8], with the neutral pions decaying into photons and the charged pions producing muons and neutrinos.

The produced proton will continue on or interact with the CMB photons again if its energy allows it. The neutron travels for long distances given the large Lorentz boost, but ultimately decays into protons, electrons, and neutrinos. The protons from this decay channel can equally interact with the CMB again, or be detected. During the processes described in eq. (2.2) and eq. (2.3) the proton looses $\sim 20\,\%$ of its energy [19], thus in each interaction with the CMB the high energy protons traveling through the universe will lose energy. This occurs until the proton looses energy to the point where the CMB photons are no longer blue shifted enough to have the energy to create a Δ^+ resonance. This limit is reached at an energy of $\sim 5 \times 10^{19}\,\text{eV}$ and is referred to as the *Greisen–Zatsepin–Kuzmin limit* (GZK

limit). Smoking gun evidence for the GZK would be a strict cutoff at this threshold energy and a CR composition which is dominated by protons. If CRs above these energies are only protons, they should be very rare and can only come from sources sufficiently close enough to meet few CMB photons in transit. This distance is called *GZK horizon*. Photons and neutrinos from the interactions and decays of particles in eq. (2.2) and eq. (2.3) should also be observed at a similar rate as the proton primaries. Ultra-high energy photons and cosmogenic neutrinos have been searched for by many experiments [20, 21, 22], however to date none have been found. Composition measurements also indicate a heavier composition at those energies instead of the lighter one required (see section 2.3).

Photo-disintegration: Another effect also considered to be part of the GZK limit as it concerns high energy particles seeing blue shifted *extragalactic background light* (EBL) and CMB photons, is photo-disintegration. In contrast to direct interactions creating new particles, in this case the photon energy is high enough to exceed the binding energy of the nucleons in the CR nuclei, which then break apart. Each interaction in this process results in two or more CRs with lower energy and lighter composition, leading to a cutoff similar to the GZK case. However, the photo-disintegration cutoff would appear more like steps in the spectrum and composition at the energies where this effect occurs for each component of the CR flux.

In truth, it is likely that all three described effects contribute to the spectrum at the highest energies. Figuring out exactly which of these effects dominates is however important to determine the sources and acceleration mechanisms of these extreme energy CRs. Because of this, a measurement of the composition of CRs at these energies is key to determining which of these processes is occurring. Unfortunately, because at these energies the flux is very low it is difficult to observe enough CRs using established composition measurement methods to infer primary mass. This means that new methods, which are able to measure the composition of these rare CRs at high statistics, are desperately needed. The AugerPrime upgrade and this thesis in particular aim to provide exactly one such methods.

2.2 Detection of Cosmic Rays

As mentioned in section 2.1.1, there are two different approaches to the detection of CRs, direct and indirect. While direct measurements have the advantage of easily reconstructing information about a CR, such as its mass composition and charge, they must be space-borne experiments or very high up in the atmosphere, as the atmosphere absorbs CRs. This means they cannot be very large, and because the flux sharply decreases with increasing

energy, this makes them only feasible up to certain energies. Thus, as energies increase to the TeV range, an alternative is needed.

2.2.1 Extensive Air Showers

While CRs themselves do not usually reach the surface of the Earth, they induce what are called *extensive air showers* (EAS) in the Earth's atmosphere, which can be measured by ground-based detectors. An EAS can generally be described as a chain reaction which creates tens of billions of particles. Through detection of these 'secondary' CRs in this cascade of particles in the atmosphere, many of the characteristics of the CR primary can be inferred. In order to do so however, one must understand how such an air shower develops in the atmosphere. A schematic view of an EAS can be seen in Fig. 2.3.

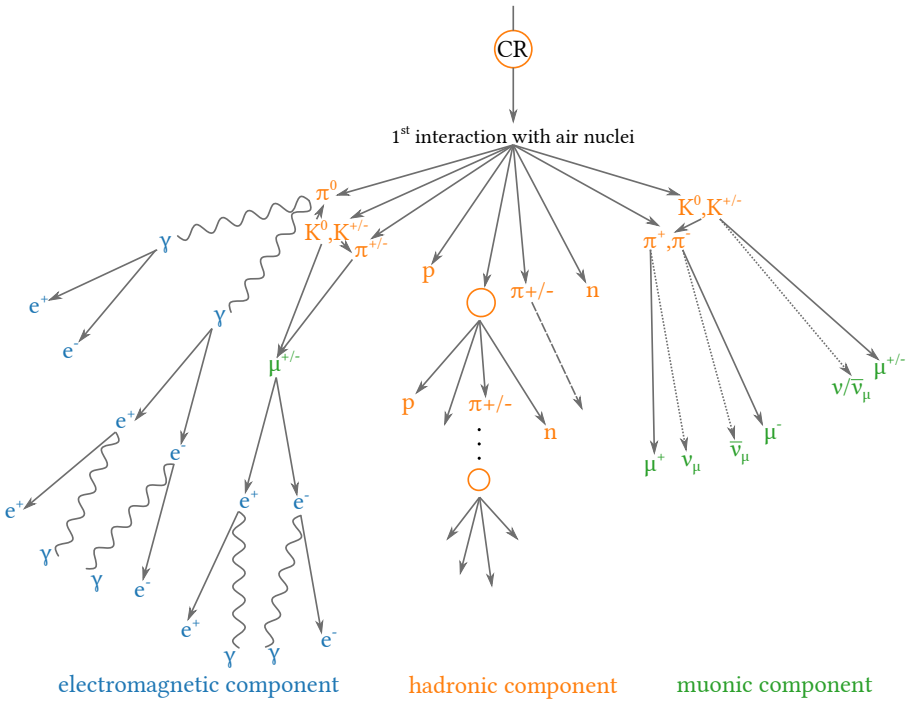


Figure 2.3: The development of an extensive air shower induced by a cosmic ray. The different types of cascades and components (electromagnetic, hadronic, muonic) are shown, indicated by the respective colors (blue, orange, green).

The Earth's atmosphere is constantly exposed to an isotropic flux of CRs which then collide with the nuclei of the atmospheric constituents (N_2, O_2, Ar, etc.). The point or depth of the CRs first interaction, X_0[1], (see section 2.2.2) with an atmospheric nucleus depends strongly on its mass and energy. In the first interaction the majority of particles produced are pions, and in smaller numbers, kaons. The three types of pions, i.e. $\pi^{\pm}$ and π^0, are produced in roughly equal amounts. The neutral pions almost exclusively decay

[1]Not to be confused with the radiation length X_0 (measured in g/cm^2) of a material, which represents the mean length to reduce the energy of an electron by a factor of $\frac{1}{e}$ in that specific material

into pairs of photons, which in turn create electrons and positrons via pair-production on atmospheric nuclei. These in turn produce more photons through bremsstrahlung causing a repeating alternation of the two processes. This part of the shower is the main contribution to what is known as the *electromagnetic component*.

The charged pions have a much longer lifetime than the neutral ones and therefore longer decay lengths. If charged pions or the charged kaons decay before interacting, they produce muons which are very likely to survive all the way to the ground. This interaction chain is referred to as the *muonic component*. If by chance the muon decays before reaching the ground it does so by producing electrons and thus contributes to the electromagnetic component.

Given their relatively long lifetimes, the high energy pions and kaons are likely to interact with other air nuclei before they decay and therefore produce more pions. Additionally, fragments of the primary particle, as well as protons and neutrons that are produced in these interactions, also interact further producing more hadronic particles. This chain is referred to as the *hadronic component*. All interactions in the hadronic component feed into both the electromagnetic and muonic component, slowly robbing it of energy [11].

As the EAS travels through the atmosphere, more and more particles are produced which share the primary energy. These secondaries will eventually reach a point where their average energy is not enough to produce further particles in interaction, and therefore the particle production rate falls below the rate at which particles are absorbed in the atmosphere. The atmospheric depth at which this turning point occurs is referred to as the depth of *shower maximum*, $X_{\max}$, since the number of particles in the shower is highest there. Due to the direct relation between number of particles and amount of emitted light, the shower is also the brightest at this point. Lastly, and most importantly for this thesis, because the number of interaction lengths between X_0 and $X_{\max}$ is dependent only on the primary energy, the location of $X_{\max}$ can be used as a proxy for X_0, which in turn provides information on the primary mass.

2.2.2 Characterizing an EAS

A simple schematic of an EAS evolving in the air can be seen in Fig. 2.4. In general, one of the most useful quantities with which to characterize a hadronic air shower is the slant depth X, which represents the amount of atmosphere the shower has penetrated up until a given point during its development. For inclined showers, those with zenith angles up to 60°, it can be easily obtained through X_v, the vertical depth. X_v is calculated by integrating the density of the air from the top of the atmosphere to the height h corresponding to the point of interest

$$X_v(h) = \int_{\infty}^{h} \rho(h')\mathrm{d}h' \,, \tag{2.4}$$

giving it a unit of g/cm^2. Then, by accounting for the zenith angle of the shower, the slant depth can be calculated as

$$X = \frac{X_v}{\cos\theta}\,. \tag{2.5}$$

The *longitudinal development profile* describes how the number of shower particles, $N(X)$, changes throughout the development of the EAS, and is illustrated in Fig. 2.4 by the orange curve above the particle cascade. Also called the shower profile, it can be well parameterized by the Gaisser-Hillas function [23]

$$N(X) = N_{\max}\left(\frac{X - X_1}{X_{\max} - X_1}\right)^{\frac{X_{\max}-X_1}{\lambda}} \exp\left(\frac{X_{\max} - X_1}{\lambda}\right), \tag{2.6}$$

where $N_{\max}$ is the maximum number of particles observed at $X_{\max}$. The parameters X_1 and λ are fit parameters which depend on primary mass and energy. As mentioned earlier, $X_{\max}$ is a variable which is strongly related to the composition of the primary CR and thus is very important to this work. Because of this, it will be described in more detail in section 3.1. The integral of this shower profile yields the total number of particles produced in the shower which is directly proportional to the calorimetric energy of the shower.

The most important geometric characteristic of an EAS is the *shower axis*, which describes the vector the shower develops along from the point of first interaction through the atmosphere to the *shower core* and is defined by its zenith, θ, and azimuth, ϕ, angle and shower core position. The term shower core (see Fig. 2.4) is used to reference both the point at which the shower axis intersects the ground and the central part of the shower immediately around the shower axis where most of the hadronic component develops. At any particular instant in time during its development, the shower would appear as a large, thin, dense disc of energetic particles called the *shower front*. The shower front is curved, as particles which travel away from the shower axis during development arrive at some observation level later in time, due to their longer path lengths. This effect also means that the shower front is relatively thin close to the shower core and widens when moving away from it, as is illustrated in Fig. 2.4.

When a shower front intersects the ground it can be characterized by the time and density of arriving particles which is referred to as the *shower footprint*. Because the shower footprint is the primary observable for the ground arrays of CR observatories, it is the shower characteristic most often used to reconstruct and classify CR air showers. The arrival time of the shower footprint at each point on the ground can be used as a means to reconstruct the shower geometry, in particular the shower axis and the curve of the shower front. With the geometry available, the density of arriving particles can be correlated to their radial distance from the shower core on the ground to obtain the *lateral distribution*

function (LDF), which can be used to reconstruct the shower energy. For this, typically the LDF is fit with an NKG-like [24, 25] function[2]

$$\rho_{\mathrm{e}}(r) = \frac{N_{\mathrm{e}}}{2\pi R_{\mathrm{M}}^2} C(s) \left(\frac{r}{R_{\mathrm{M}}}\right)^{(s-2)} \left(\frac{r}{R_{\mathrm{M}}} + 1\right)^{(s-4.5)} , \tag{2.7}$$

with parameters

$$s = \frac{3}{1 + 2X_{\mathrm{max}}/X} , \quad \text{and} \quad R_{\mathrm{M}} = 0.0265 X_0 (Z + 1.2) . \tag{2.8}$$

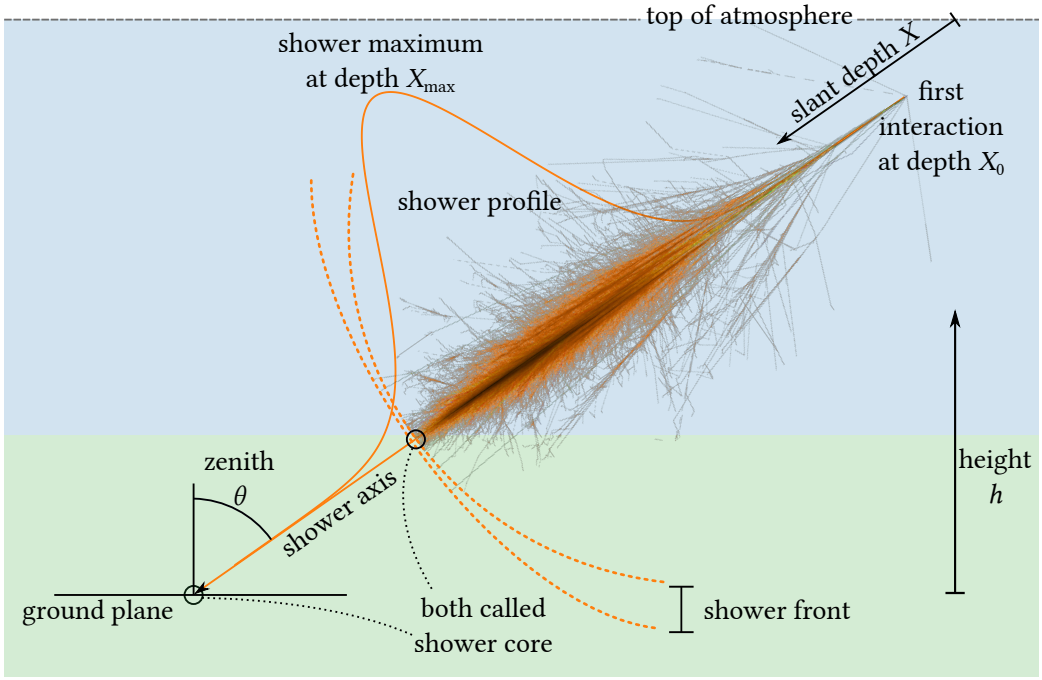


Figure 2.4: Schematic depicting an EAS.

The parameterized LDF is then zenith corrected and converted to the energy of the primary particle using either models or cross calibration from atmospheric measurements of the shower profile (for more details see chapter 3.) Of particular importance to this thesis is the fact that each of the three components of an EAS evolves differently, has different lateral particle density distributions, and therefore different shower footprints on the ground. Since the relative fraction of particles in each component depends on the primary composition, if the footprints of the electromagnetic, muonic, and hadronic showers can be distinguished, the type of particle which started the EAS could be reconstructed by a ground array.

[2]The NKG (Nishimura-Kamata-Greisen) function was developed as a theoretical approach to describe the lateral distribution of electrons and muons.

2.3 The Composition of Ultra-High Energy Cosmic Rays

As mentioned previously in this chapter, the composition of CRs is very important to understand more about the sources of CRs and their acceleration mechanisms. While at lower energies, with space-borne detectors, the composition of detected cosmic rays can be measured directly and is described in section 2.1.1, this is not possible anymore above a few TeV as CRs are detected indirectly via their EAS. With indirect detection, composition measurements become model dependent, since the interaction processes taking place in an EAS must be simulated and therefore rely on different models, e.g. [26, 27, 28]. This introduces uncertainties to measurements of the primary mass as these models are based on the extrapolation of accelerator measurements to the much higher energies of CRs.

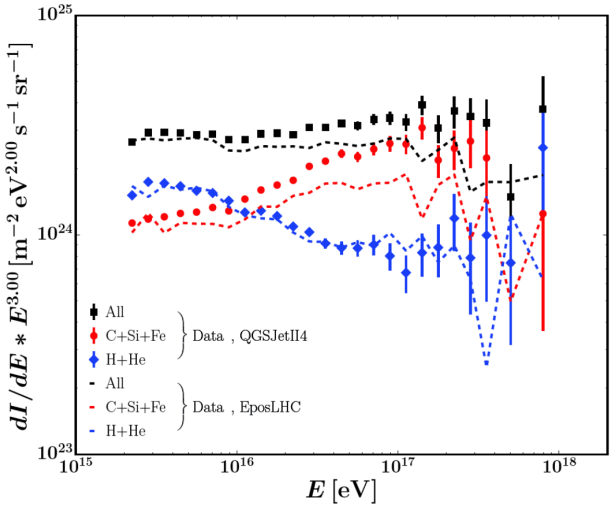


Figure 2.5: CR spectrum as measured by KASCADE+KASCADE-Grande. The black squares indicate the measured all-particle spectrum, while the red circles and blue diamonds represent the heavier and lighter mass components, respectively. Taken from [29].

One approach to indirect composition measurements is to infer the mass of primary CRs from the differences and ratios of the three shower components measured in the detected particle shower. This method was leveraged by the *KArlsruhe Shower Core and Array DEtector* (KASCADE) experiment [30], which measured the electromagnetic and muonic components separately, as well as at its extension KASCADE-Grande, which measured all charged particles. They used a combination of different detectors that are each dominantly sensitive to one of the shower components of an air shower to reconstruct the number of particles $N_{e/\gamma}$, N_{μ}, and N_{ch} from the different detector responses. The ratios between the electromagnetic and muonic component $N_{e/\gamma}/N_{\mu}$ (KASCADE) and the ratio between the charged and muonic component N_{ch}/N_{μ} (KASCADE-Grande) are sensitive to different primary masses. This is because primaries with the same $N_{e/\gamma}$ or N_{ch}, will have different

N_μ. Also shower-to-shower fluctuations for events of the same primary mass tend to fluctuate similarly in N_{ch} or $N_{e/\gamma}$ and thus primary composition is reconstructable on an event-by-event basis. A parameter k is calculated for different zenith intervals as

$$k = \frac{\log_{10}(N_{ch}/N_\mu)_{event} - \log_{10}(N_{ch}/N_\mu)_H}{\log_{10}(N_{ch}/N_\mu)_{Fe} - \log_{10}(N_{ch}/N_\mu)_H}, \tag{2.9}$$

where $\log_{10}(N_{ch}/N_\mu)_H$ and $\log_{10}(N_{ch}/N_\mu)_{Fe}$ are calculated using Monte Carlo simulation and therefore depend on the hadronic interaction model chosen. Per definition, k is centered around one if the primary is iron and centered around zero if the primary is a proton. Each event can be marked as lighter (H+He) or heavier (C+Si+Fe) in composition, based on the value of k for that event.

Composition results for combined KASCADE and KASCADE-Grande data using this method can be seen for two different hadronic interaction models in Fig. 2.5. The results verify the first knee structure mentioned in section 2.1 at 3×10^{15} eV for the lighter elements, followed by increasing dominance of heavier elements and a second knee at 10^{17} eV for the heavier elements. Additionally, an increase in lighter elements after the second knee is visible [29].

Another approach for measuring the composition of CRs is to use the longitudinal parameter X_{max} mentioned in section 2.2.2. This has been most notably used by the Fly's Eye experiment, the Pierre Auger Observatory, and the Telescope Array [31, 12, 13, 32]. Shower-to-shower fluctuations in the first few hadronic processes of the cascade mean that the primaries mass can only be inferred from the distribution of shower maxima, instead of on an event-by-event basis.

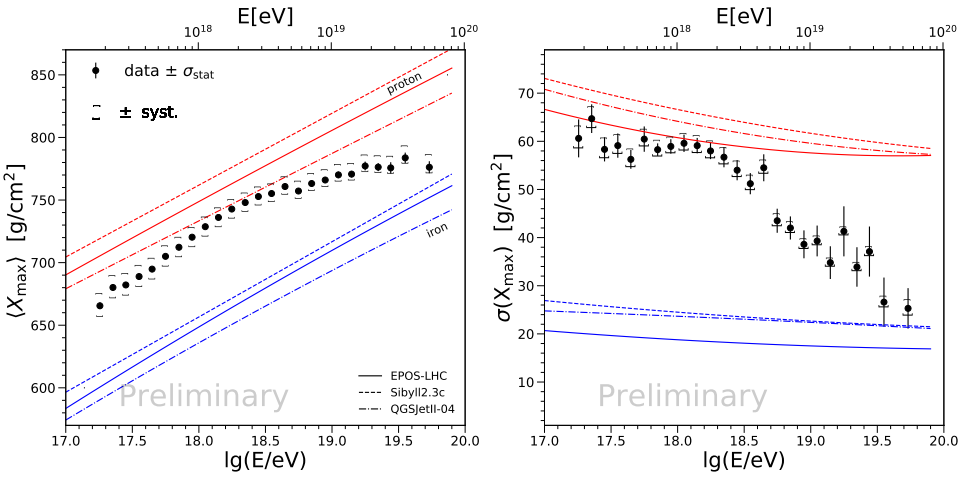


Figure 2.6: Pierre Auger Observatory measurements of $\langle X_{max} \rangle$ (left) and $\sigma^2(X_{max})$ (right), showing X_{max} getting deeper towards the ankle indicating light composition and then shallow beyond the ankle indicating heavier composition. Taken from [14].

Assuming a mixed composition, this distribution is a superposition of all the individual

distributions $f_i(X_{\max})$ created by the different primary particles i with mass A_i multiplied by the fraction of the flux represented by that primary type p_i:

$$f(X_{\max}) = \sum_i p_i f_i(X_{\max}) \tag{2.10}$$

This first two moments of this distribution, i.e. its mean $\langle X_{\max} \rangle$ and variance $\sigma^2(X_{\max})$, respectively, are linearly proportional to the logarithm of the primaries mass $\ln(A)$ [12] and its variance. A detailed review on the proportionality factors between $\ln(A)$ and $\langle X_{\max} \rangle$ and $\sigma^2(X_{\max})$ is given in [33].

Recent results on $\langle X_{\max} \rangle$ and $\sigma^2(X_{\max})$ from the Pierre Auger Observatory [14] can be seen in Fig. 2.6, as well as the inferred logarithm of the primary mass depending on the hadronic interaction models used in Fig. 2.7. In accordance with the KASCADE results, the composition gets lighter towards the ankle, where it clearly has a significant proton fraction. As energies increase beyond the ankle, the composition gets increasingly heavier again. A limiting factor in these results and others at the same energies, and as a result in answering the earlier mentioned open questions, is a lack of statistics at the highest energies. The necessity of increasing the high energy statistics to answer these open questions is one of the driving factors behind the upgrade of the Pierre Auger Observatory, AugerPrime, and is the focus of this thesis. The goals of AugerPrime are to shed light on the mass composition and source of the flux suppression of UHECRs, as well as better understanding the hadronic interactions in EAS at these energies well beyond those available at collider experiments.

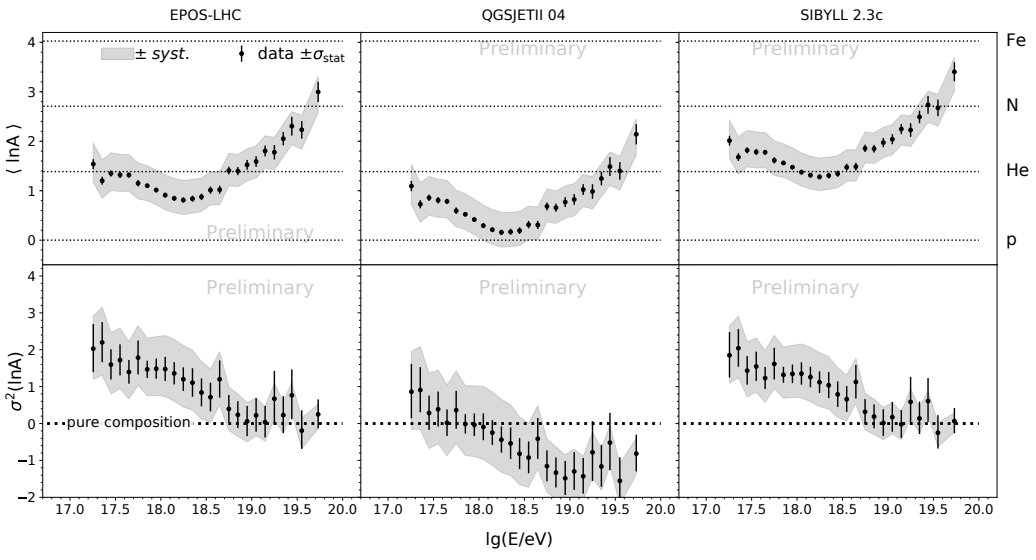


Figure 2.7: Measurements of $\ln(A)$ moments performed at the Pierre Auger Observatory also indicating lighter composition around the ankle and heavier beyond. Here, it becomes apparent that the results are very model-dependent. In particular QGSJETII-04 appears non-physical above 1 EeV due to its purer than pure composition. Taken from [14].

3
The Pierre Auger Observatory

> *Instruments only register things they are designed to register. Space still contains infinite unknowns*
>
> — Mr. Spock, *Star Trek: The Original Series*

The Pierre Auger Observatory is the world's largest detector for measuring the extensive air showers produced by cosmic rays. It is located in the Pampa Amarilla near the city of Malargüe in the province of Mendoza, Argentina [34]. It was completed in 2008 and covers a surface area of around 3000 km^2.

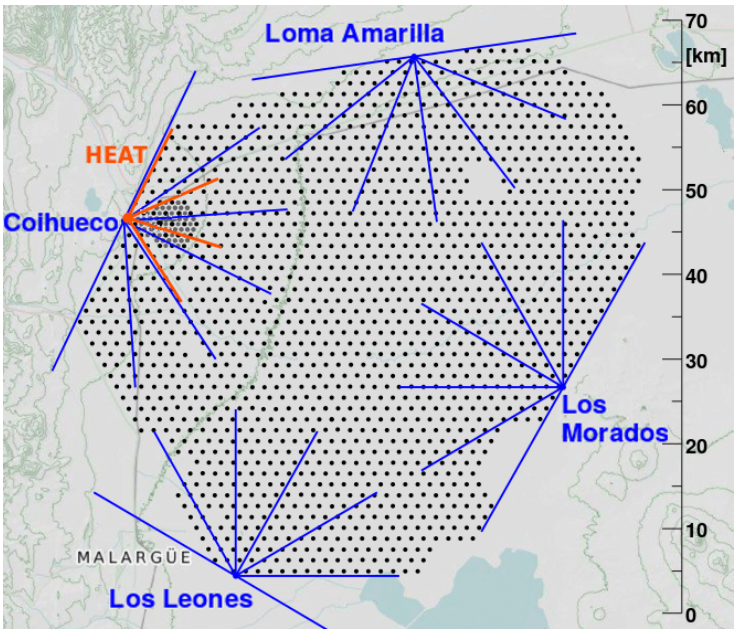


Figure 3.1: Layout of the Pierre Auger Observatory in Argentina. Each black dot represents a WCD, while the lines indicate the field of view of each telescope at the different FD sites.

There are two main detectors used at the Observatory, the *Fluorescence Detector* (FD) and the *Surface Detector* (SD). The SD and FD will be covered in this thesis in further detail. Information on other detector components such as the *Radio Detector* (RD) or the *Underground Muon Detector* (UMD) can be found in [35, 36, 37] and [38, 39], respectively. These different detector types are operated as a hybrid system to seize the advantages of each detector type, making the Pierre Auger Observatory a hybrid Observatory. The largest part of the Observatory in terms of exposure is the SD which consists of 1660

water Cherenkov detectors (WCD). The atmosphere above the SD is monitored by the 27 telescopes of FD, with six telescopes at each of the four FD building sites and an additional three in the low energy extension *high elevation fluorescence telescopes* (HEAT) [40]. All these components can be seen in Fig. 3.1. With these two main detectors, a complimentary measurement of the shower development in the atmosphere, via the FD, as well as its signal on the ground, via the SD, is possible, forming the heart of the hybrid detector [34]. The RD, part of the ongoing Observatory upgrade, AugerPrime, is based on the Auger Engineering Radio Array (AERA) and will add another high statistics detection method to events with zenith angles above 60°.

3.1 The Fluorescence Detector

The charged particles created in an EAS will excite nitrogen molecules in the Earth's atmosphere as they pass. These will then emit fluorescence light in the UV range, i.e. in the wavelength range 300 nm to 430 nm. The number of fluorescence photons emitted is directly proportional to the electromagnetic energy lost by the charged particles of the EAS and therefore also the total number of particles in the EAS. Because of this, by detecting the fluorescence emission throughout the EAS the FD directly measures its energy deposit as a function of its atmospheric depth, $\frac{dE}{dX}$, also known as the longitudinal development profile of the shower. The integral of the profile

$$E_{\mathrm{cal}} = \int \frac{dE}{dX} dX \tag{3.1}$$

is directly proportional to the electromagnetic energy of the EAS, which comprises $\sim 90\,\%$ of the primary particle's energy, thus providing a nearly calorimetric measurement [41]. The depth of shower maximum, X_{max}, is also easily inferred from the FD measurement and because it is roughly proportional to the logarithm of the primary mass, $\ln(A)$, the FD is sensitive to the composition of the primary. Because the FD directly measures X_{max}, it currently has the best resolution for shower-by-shower composition measurements among the detectors at the Observatory and in general.

The instrumentation of the FD primarily consists of 24 telescopes overlooking the array housed in four different FD sites or *Eyes*: Coihueco, Los Morados, Loma Amarilla and Los Leones, indicated in Fig. 3.1. At one of these (Coihueco) the low energy extension HEAT observess the upper atmosphere directly above a 66-detector sub-array of the SD. In this 23.5 km^2 region the reduced 750 m spacing of the detector stations and the higher observation level of HEAT allow for measurements of events down to nearly PeV energies. Each fluorescence telescope essentially consists of a UV filter window, a 13 m^2 segmented spherical mirror and a camera which consists of 440 *photo-multiplier tubes* (PMTs). The telescope is designed to give the camera a fixed 30°×30° field of view (FoV), which results

in each PMT observing a particular $\sim 1.5°$ section of the sky.

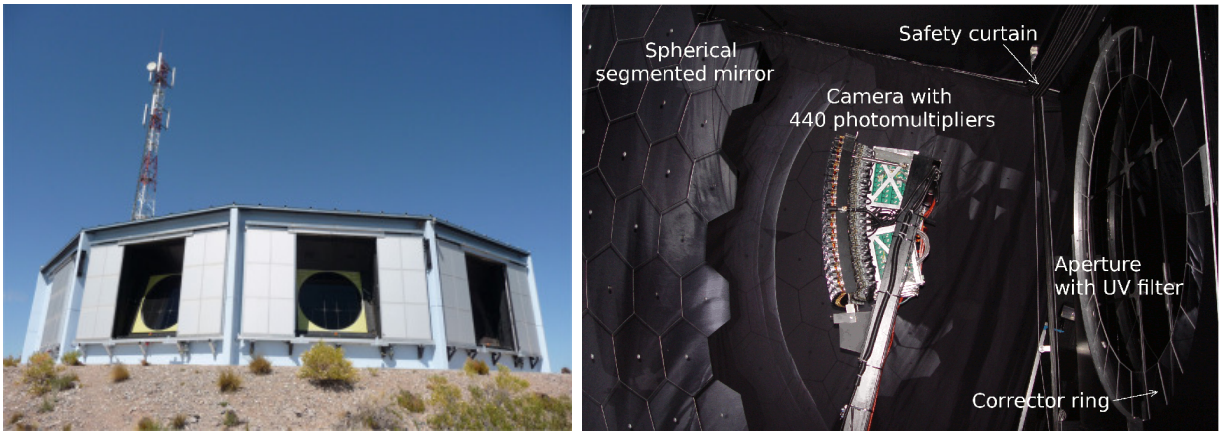


Figure 3.2: Left: FD building from the outside, with three open telescopes visible. Right: Inside view of an FD telescope with the main components labeled. From [34].

During an FD measurement, first, most background light is removed from the UV fluorescence light of the shower via the filter window. The light is then gathered by the mirror and reflected onto the PMT camera, which sits in the focal plane of the mirror. Each PMT then records the UV light within its FoV with a 100 ns timing resolution in an intensity and timing trace of the shower. From this the shower geometry and profile can be reconstructed [41].

The FD requires very dark, cloudless nights to make a measurement. Because of this, it has a low duty cycle of 14–15 %, resulting in relatively poor event statistics at the highest energies. As the duty cycle of the SD (see section 3.2) is much higher than the FDs, the SD collects much higher event statistics at the highest energies. The composition resolution of SD-only analyses however is not as good as that of the FD. Because of this, an update to the SD to enhance its shower-by-shower composition measurements capabilities, such as the currently on-going upgrade AugerPrime, has the potential to greatly increase composition information at the highest energies. Development of a technique to achieve this is the motivation of this thesis.

3.2 The Surface Detector

As mentioned before, the standard SD primarily consists of 1600 WCDs. These WCDs are placed in a hexagonal grid with a spacing of 1500 m between stations. Each WCD consists of a plastic tank with a diameter of 3.6 m, which is filled with 12 000 ℓ of highly purified water. Three PMTs, of 9 inch diameter each, monitor this water to measure the Cherenkov light produced when the particles of an air shower pass through the water [34]. Each of those PMTs is placed 1.2 m from the center of the water tank, looking downwards in a symmetric layout. The water itself is held inside an opaque, highly reflective liner

composed of Tyvek©. The high purity water is necessary to achieve the lowest possible attenuation for Cherenkov light, while also maintaining the stability of the water and liner properties over the long operation time of the Observatory.

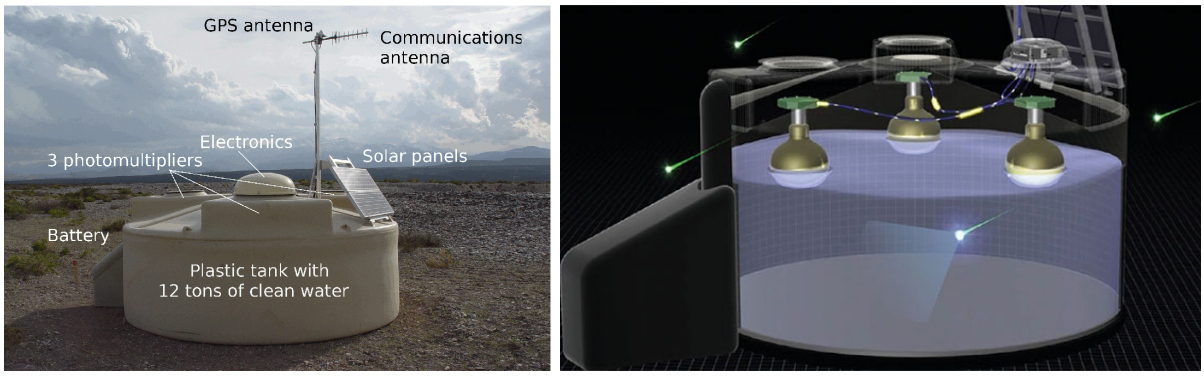


Figure 3.3: Left: Example of a WCD in the array with the different components marked. Right: An inside view illustration of such a WCD with the PMTs shown and a light producing particle going through. From [34].

Each of these SD stations is fully self-sufficient and autonomous, via its GPS receiver, radio communications antenna, micro-controller, and a solar panel with a battery, which supply the 10 Watts of power needed by the electronics and PMTs. The full setup of such a station can be seen in Fig. 3.3. Because each SD station is relatively unaffected by the environment, the SD has a duty cycle of nearly 100 % and constantly records the intensity and timing of particle flashes passing through it.

3.2.1 The SD Signal Calibration

The electronics of the SD consist of six 40 MHz *flash analog-to-digital converters* (FADCs), which digitize the signals of the three PMTs in both a *low gain* (LG) channel and a *high gain* (HG) channel which is amplified by a factor of 32 [34]. These two channels together provide enough dynamic range to measure the flux of particles well, both near the shower core ($\sim 1000\,\text{particle(s)}/\mu\text{s}$) and far from it ($\sim 1\,\text{particle(s)}/\mu\text{s}$). For both of these FADC channels, a signal trace is recorded with a length of 768 bins for each PMT. Due to the above mentioned sampling rate of 40 MHz, each bin has a time width of 25 ns.

The signal of each detector station differs and therefore has to be converted to units of a *vertical-equivalent muon* (VEM). One VEM is defined as the charge deposited in that station by one *vertical and central through-going muon* (VCT) muon. A VEM serves as the unit of measurement for the signal created by particles passing through the stations, and is needed in order to have a universal measurement standard for all stations. While the SD is not able to filter out and measure only VCTs, calibration is possible by using the response of the PMTs to random atmospheric muons in combination with reference measurements from a muon telescope which precisely measures VCTs.

To perform the SD calibration mainly two calibration histograms are used, the charge histogram (for each PMT and their sum) and the pulse height histogram (for each PMT and their sum). The charge that corresponds to a VEM, is obtained by integrating the amplitudes in time over the signal pulse duration. As can be seen in Fig. 3.4, the Cherenkov light created by atmospheric muons produces a peak in the charge histogram, which, with enough events, always corresponds to $Q_{VEM}^{peak} \sim 1.09\,\text{VEM}$. This peak is used together with the reference measurement of $Q_{VEM} = 1\,\text{VEM}$ of the muon telescope (also shown in Fig. 3.4) to obtain the calibration value needed to scale Q_{VEM}^{peak} for the sum of all three PMTs and thereby express the charge in VEM [42]. Each of these is created every minute, which means that in the case of an event trigger in the array, there is calibration data available from the previous minute, guaranteeing high calibration accuracy.

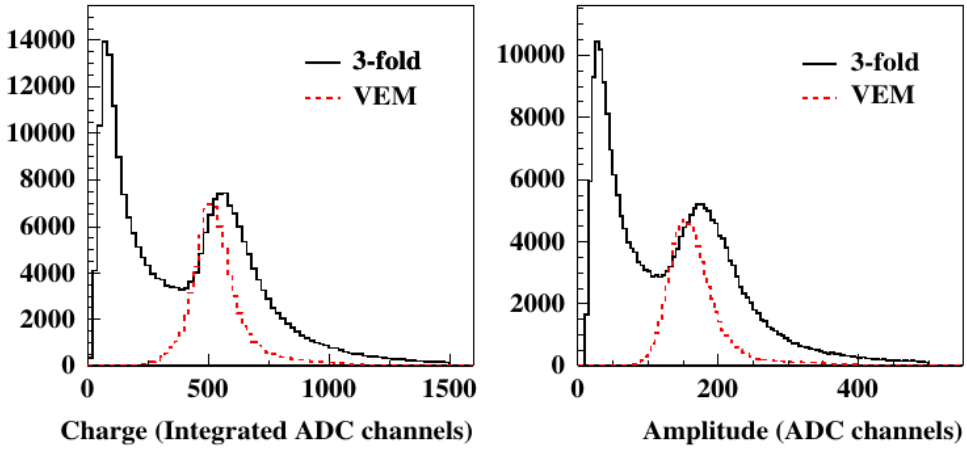


Figure 3.4: Left:The charge histogram (black), is the sum of the responses of all three PMTs to background noise from the last 60 s, while the dashed red line represents the reference measurement done by an external muon telescope. The first peak in the charge histogram is noise from low-energy particles while the second peak is used to calibrate the unit of VEM in the corresponding station. Right: The pulse height histogram (black) that is used to calibrate I_{VEM}^{peak} to be used for trigger thresholds. This is again done with the red dashed line from the muon telescope measurement as reference. Taken from [42].

All triggered events of the SD require the signal trace, a measure of the current seen by the PMT which is expressed in ADC counts, to be over a certain threshold. Therefore, the station current must also be calibrated to a reference unit I_{VEM}. To do this, the same technique as is used for the VEM is employed. Atmospheric muons will also produce a peak current I_{VEM}^{peak} in the pulse height histogram. I_{VEM}^{peak} is then also calibrated via the reference current from the muon telescope, I_{VEM} [42].

3.2.2 The SD Trigger System

Overall, there are three main trigger functions implemented in the front-end electronics of each station. These are the scaler trigger, calibration trigger and main shower trigger.

While the scaler trigger, which records pulses with a very low signal threshold, is useful for performing complementary physics such as space weather [43], the calibration trigger is crucial for operating the detector. The calibration trigger is set so that it measures low pulses from each PMT for 20 bins as soon as any bin sees a current greater than $0.1\,\mathrm{I}_{\mathrm{VEM}}^{\mathrm{peak}}$. This data is then used to perform the calibration processes described earlier.

The most important trigger is the main shower trigger. Because of the very high rate of random accidental muons and the very low rate of UHECRs, obtaining every single measurements from every station is decidedly impractical. This makes it necessary to define a set of conditions that need to be fulfilled in order for an event to be classified as coming from an UHECR. This trigger system is hierarchical and has five levels, T1 through T5. The first three levels provide the foundation of all later trigger levels and are illustrated in Fig. 3.5.

The first level, T1, consists of two triggers, a simple threshold (TH) trigger, which is activated if all three PMTs record a signal above $1.75\,\mathrm{I}_{\mathrm{VEM}}^{\mathrm{peak}}$, and a time-over-threshold (ToT) trigger. The ToT requires a signal over a threshold of $0.2\,\mathrm{I}_{\mathrm{VEM}}^{\mathrm{peak}}$ in at least two PMTs in a time window of 120 bins which lasts for a minimum of 13 bins. The T1-TH is especially targeted at identifying the larger, inclined events, where the muonic component dominates. The T1-ToT is targeted at smaller, more vertical events, low-energy showers close to the shower core, and high-energy showers far from it. In 2013, two additional trigger methods, ToTd and MoPS, were added to reduce the influence of single muons and lower the energy threshold of the array and are described in more detail in [34].

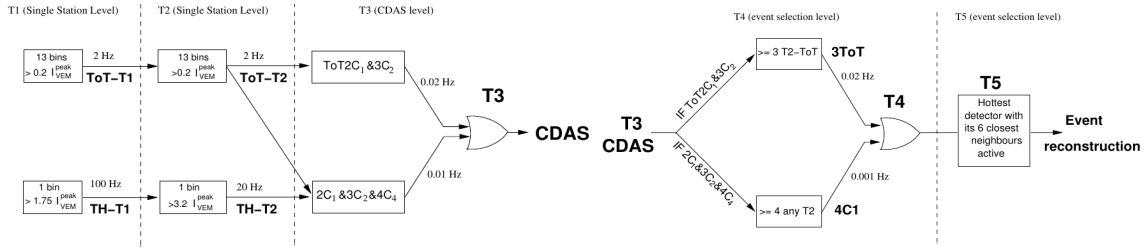


Figure 3.5: Left: Schematic overview showing the low-level trigger hierarchy, that events need to pass to be send to CDAS. Right: Schematic overview showing the high-level trigger hierarchy, that is needed to differential air shower events from accidental trigger by e.g. atmospheric muons. Taken from [44].

For the second level, T2, the previously identified T1-ToT triggers are always promoted to T2-ToT, while the T1-TH triggers have to fulfill an additional criterion of a 3-fold PMT coincidence with each PMT being over a signal threshold of $3.2\,\mathrm{I}_{\mathrm{VEM}}^{\mathrm{peak}}$. All T2 triggers are sent with their timestamp via wireless communication to the *Central Data Acquisition System* (CDAS) (see Fig. 3.5) to be combined with the signals from others detectors to form an event. The first step towards forming an event is the third level, T3, in the trigger hierarchy. It is again split into two different modes, both of which involve the fulfillment

of many spatial and temporal conditions. Examples of both modes can be seen in Fig. 3.6.

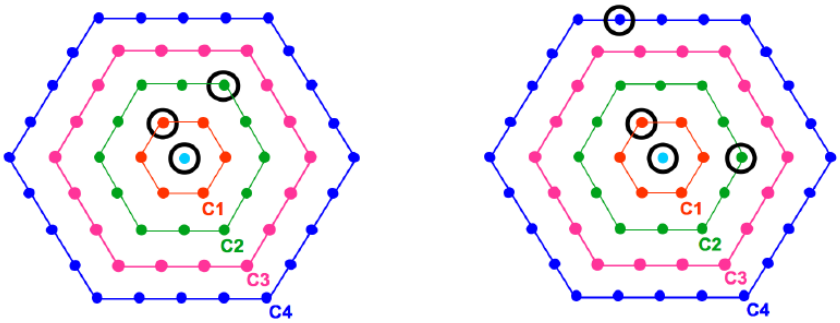


Figure 3.6: The two different types of T3 trigger. Example configurations of triggered central stations and coincidentally triggering stations in the surrounding hexagons are circled. Taken from [44].

T3 is based on coincident signals of a station and its surrounding hexagons, with C_n used in the naming scheme defining the n^{th} set of neighbors, i.e. the order of surrounding hexagons. For the ToT2C_1&3C_2 trigger mode at least three participating detectors need to have passed the T2-ToT trigger. An additional requirement of compactness (see Fig. 3.6 left) is also imposed. The $2C_1$ in the name references that at least one station needs to be within the C_1 hexagon and likewise $3C_2$ indicates for the third station to be at least as close as the C_2 hexagon. This trigger also has a temporal requirement which demands that the T2 triggers of the other stations arrive within $(6 + 5C_\mathrm{n})\,\mu\mathrm{s}$ of the first one [44]. The second T3 builds on this but is of a less restrictive nature. All stations must have either ToT-T2 or TH-T2 triggers (see Fig. 3.5) and a 4-fold coincidence is required. The first two neighbors of a selected station must be in the same order as for the ToT2C_1&3C_2, whereas the fourth station can be as far away as the 4^{th} order. This trigger is accordingly named $2C_1$&3C_2&4C_4 and is especially targeted at inclined showers, with a larger shower footprint on the ground.

After these steps, further event selection is done according to the trigger levels T4 and T5, as displayed in Fig. 3.5. Not all the triggers that pass the T3 level are real air shower events, so a physics trigger, T4, is needed to select only those events likely caused by EAS. There are again two different sets of T4 triggers. The first one, 3ToT, requires the arrival times in at least three additional stations in a triangle, that have passed the T2-ToT, to fit to a planar wave front traveling at the speed of light. This is particularly effective at picking out events with a zenith angle of less than 60°. The second type of T4 trigger, 4C1, adds to that by selecting events with four or more stations, that have passed any kind of T2 level trigger and can also be fit with a planar wave traveling at the speed of light.

Finally, the fiducial trigger, T5, removes events which are close to the borders of the array and thus may have missing stations in the footprint, which could be leading to wrong reconstructions. To achieve this, T5 only selects an event if the station with the

highest signal has all its 6 closest surrounding neighbors, i.e. C_1, working at the time of measurement. This selections ensures that high energy events that fall close to the border, but inside the array, may still be kept and reconstructed while limiting potential biases from incomplete events [44].

3.2.3 The SD Shower Reconstruction

An in-depth overview on the SD shower reconstruction is given in [45]. The most important step for the analyses presented in this thesis are the reconstruction of the shower geometry and the shower energy. First, the arrival direction of the shower is approximated assuming a spherical shower front

$$c(t_{\mathrm{sh},i} - t_0) = |\vec{x}_0 - \vec{x}_i| \ , \tag{3.2}$$

with the speed of light c. Here, $t_{\mathrm{sh},i}$ is the arrival time of the shower front at a station i with position $\vec{x}_i$. The parameters t_0 and $\vec{x}_0$ are the start time and position of the shower. The arrival direction obtained this way can then be used to calculate the shower core position $\vec{x}_{\mathrm{c}}$ as well as the shower size by using the log likelihood method and simultaneously fitting the LDF of the form

$$S(r) = S\left(r_{\mathrm{opt}}\right) \left(\frac{r}{r_{\mathrm{opt}}}\right)^{\beta} \left(\frac{r + r_s}{r_{\mathrm{opt}} + r_s}\right)^{\beta+\gamma} \tag{3.3}$$

where S is the signal at the perpendicular distance to the shower axis r. The parameter r_s is fixed to r_s=700 m and r_{opt} denotes the optimum distance based on the array spacing. For the SD array with the spacing of 1500 m, this amounts to r_{opt} ≈1000 m. Once the position of the shower core $\vec{x_{\mathrm{c}}}$ is found, the shower axis $\vec{a}$ can be determined as

$$\vec{a} = \frac{\vec{x}_0 - \vec{x}_{\mathrm{c}}}{|\vec{x}_0 - \vec{x}_{\mathrm{c}}|} \, . \tag{3.4}$$

The parameter $S(1000)$ is an estimator of the shower size and thus related to the primary energy of the EAS. To reconstruct the primary energy from $S(1000)$, atmospheric attenuation has to be considered. With increasing zenith angle the slant depth to ground of the SD increases. As a result, very inclined showers are overall detected at a later shower "age" and are subject to stronger particle attenuation and geometric effects. This leads to the shower size $S(1000)$ having zenith dependence. To correct for this effect a constant intensity cut (CIC) method is used described in detail in [23, 46]. This results in an attenuation function that can be used to convert $S(1000)$ to a zenith independent shower size. Assuming that the arrival rate of cosmic rays above a given energy does not depend on zenith angle, a minimum $S(1000)_{\mathrm{min}}$ should exist above which the arrival rate of a larger shower size should be equal to a constant at a given zenith angle θ. Once this attenuation

function $f_{\mathrm{CIC}}(\theta)$ is found, it can be used to convert $S(1000)$ via

$$S_{38} = \frac{S(1000)}{f_{\mathrm{CIC}}(\theta)}\,, \tag{3.5}$$

where 38 refers to the median zenith angle of all events of $\overline{\theta} = 38°$. This transforms $S(1000)$ into the equivalent size the shower would have had, had the primary particle arrived with a zenith angle of 38°. Using hybrid events and the nearly-calorimetric energy measurement of the FD, this parameter can then be calibrated [47] by fitting with a power-law calibration curve

$$E_{\mathrm{FD}} = A\left(\frac{S_{38}}{\mathrm{VEM}}\right)^{B}\,. \tag{3.6}$$

Using A and B from this fit an energy can be assigned to every SD event

$$E_{\mathrm{SD}} = A\left(\frac{S(1000)}{f_{\mathrm{CIC}}(\theta)\times\mathrm{VEM}}\right)^{B}\,, \tag{3.7}$$

also referred to in this document as $E_{\mathrm{SD,rec}}$.

3.3 The AugerPrime Upgrade

The AugerPrime upgrade of the Pierre Auger Observatory was conceived in 2015 [39], with the construction of the first prototypes starting in 2016. It is primarily an improvement and extension of the detector system and consists of five main aspects:

1. The installation of plastic scintillators on top of the existing WCDs, which provide a complementary measurement of air showers. A *surface scintillator detector* (SSD) is more sensitive to the electromagnetic component of an EAS, while the WCD has a stronger response to the muonic component. These differing responses between the two detectors will enhance the precision of primary particle composition measurements made by the SD on an event-by-event basis. The SSDs are also easily deployable, which is important considering the full array size [48].

2. The replacement of the SD electronics, specifically the *unified board* (UB) and the separate SD front-end board, with an upgraded version, the *upgraded unified board* (UUB) that combines the two. The UUB is a significant improvement because it is capable of processing both the WCD and SSD signals and has a faster sampling rate, better timing accuracy, and higher dynamic range. Additionally, a fourth *small PMT* (sPMT), is being installed inside the WCD to extend its dynamic range [49, 50].

3. The increase of the FDs duty cycle by $\sim 50\,\%$ by extending its observation time to periods with higher night sky background.

4. The deployment of an *underground muon detector* (UMD) in the $23.5\,\mathrm{km}^2$ sub-array of the SD (750 m), to provide direct muon measurements. This can be seen as an extension of the currently used AMIGA underground muon detectors.

5. The deployment of radio antennas on top of each SD station [37]. This enables complementary radio measurements of air showers. It serves a similar purpose as the SSD, but in the very inclined zenith range (65° to 85°), whereas the SSD is sensitive up to $\sim 60°$.

In the context of this thesis, the most important parts of this upgrade are the addition of the SSD and the electronics upgrade.

After the extended operation of the Pierre Auger Observatory until the 2030s, the event statistics will have more than doubled. Furthermore, with the upgrade all events taken will have additional mass sensitivity compared to the existing data set. As mentioned earlier, this additional information can offer an improved understanding of the composition of primary particles at the highest energies [39]. This could also lead to a reduction in systematic uncertainties related to modeling hadronic showers via Monte Carlo simulations, by providing a better understanding of the particle contents of showers at energies well above what is currently available at colliders. The knowledge gained from this could be used to reanalyze old data, especially in the context of improving energy reconstructions in regards to the current muon deficit, where the muon content in EAS simulations is much lower than the one detected in data [51]. Lastly, it should help in photon and neutrino searches, through increased sensitivity.

3.3.1 The Surface Scintillator Detector

The SSD was chosen with the goal of providing a different detector response than the WCD, while keeping the need for robustness and minimal maintenance in mind. A schematic of the resulting design of the SSD is shown in Fig. 3.7, as well as an example of one of the upgraded stations (without the RD) from the AugerPrime *Engineering Array* (EA).

Each SSD consists of two scintillator parts with a combined area of $\sim 3.8\,\mathrm{m}^2$ [48]. The parts consist of 12 bars made from extruded polystyrene with emission lines between 330 nm to 480 nm. The light in the scintillator bars is collected by wavelength-shifting fibers. Their absorption spectrum matches the emission of the scintillator material, while their own emission spectrum is above 450 nm. Special focus was put on the routing of these fibers and guiding them to the front of the PMT to ensure low light loss. For this, each fiber starts and ends at the *cookie*, which bundles all fiber ends in front of the PMT. The fibers themselves are passed through the scintillator bars by guiding them in U-turns

from one bar to another before returning to the cookie. From the cookie to the start of a scintillator bar the light needs to travel $\approx 1.1\,\mathrm{m}$, which results in only photons with a wavelength above 500 nm surviving due to wavelength shifting and attenuation [48].

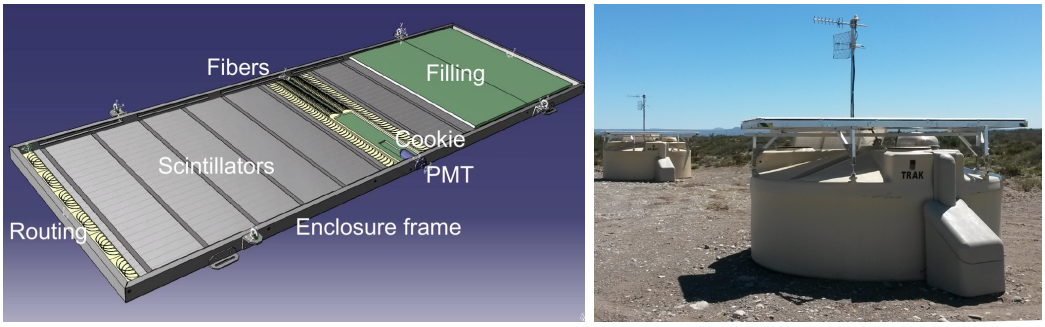


Figure 3.7: Schematic view of the scintillator plane and its components (left) and assembled upgraded station in the Engineering Array (right).

The overall shift to larger wavelengths is important, because the quantum efficiency of the PMTs selected for the SSD is higher for these larger wavelengths than for those of the light emitted directly from the scintillator. The PMT chosen for the upgrade is the model R9420 manufactured by Hamamatsu, which has a 1.5 inch bi-alkali photocathode specifically suited for scintillator measurements [48]. Its high voltage power supply is soldered to the PMT based on a custom-made design by the ISEG company. The calibration of this PMT is carried out much in the same way as those in the WCDs. The only major difference is the calibration unit involved, as the scintillator is more sensitive to electromagnetic particles rather than muons. Hence, the calibration unit of the SSD is the number of *minimum ionizing particles* (MIP). Further details and example charge histograms are provided in section 4.7. The triggering of an upgraded station does not differ much from the WCD-only trigger, as the SSD is only operated in a passive mode, meaning that the SSD data is read out, if the WCD triggers. Adjustments to the existing triggers are only necessary in regards to the change in the sampling rate.

3.3.2 The Upgraded Unified Board

The UUB was designed to improve on a few aspects with respect to the original UB in combination with the SD front-end board. These improvements include better data quality, timing accuracy, and an increased dynamic range [49]. The data quality is improved by using a higher sampling rate of 120 MHz as compared to the 40 MHz of the UB, resulting in signal traces with shorter time bins of 8.33 ns. It is important that backwards compatibility with the current WCD-only data-sets is maintained and so the total timespan of the SD traces is kept nearly the same. The increased sampling rate means that while the UB reads out the WCDs signals with 768 time bins of 25 ns, the UUB reads signals out with 2048 bins. This means that, while the current triggers must be adapted

to the higher sampling rate, it is also possible to use them as is on the upgraded board by down-sampling [49]. All the data in chapter 4 has been obtained with this down-sampling method. The improvement in detector timing is achieved by the higher sampling rate, which gives a more precise measurement on the start time of the signal trace, as well as a new GPS module. For more information on the timing see section 4.5.

The extended dynamic range, matched between SSD and WCD, will specifically allow for measuring signals at distances as close as 250 m from the shower core without saturating [50]. The dynamic range in the SSD is similar to that described in section 3.2.2, as it has a HG and an LG channel. In the SSD, the HG signal is amplified by a factor of 32 while the LG is attenuated by a factor of 4. The effect of these gain settings is further studied in section 4.3. To achieve a comparable dynamic range in the WCD a hardware extension is required which is, as mentioned above, achieved by the addition of the sPMT [50].

3.3.3 Deployment of AugerPrime

The deployment of SSDs began with the assembly of the AugerPrime EA, which has been taking data in the array since October 2016

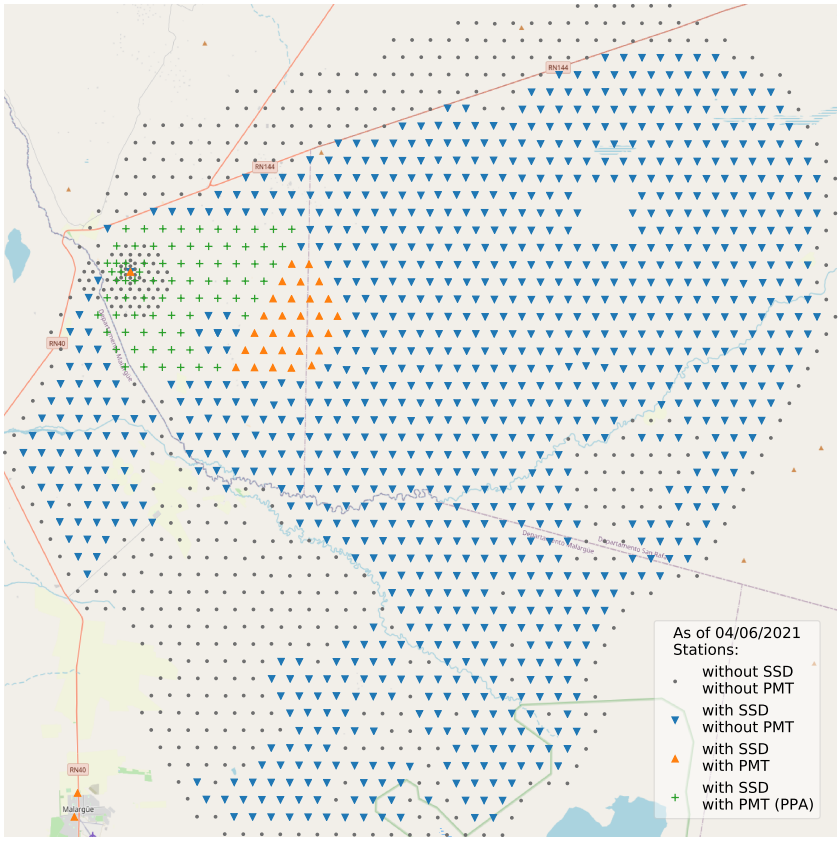


Figure 3.8: Overview of the deployment (status 07.04.2021). Non-upgraded stations are shown as circles (gray), stations with SSDs but without PMT as down-triangles (blue), stations with both an SSD and PMT as up-triangles (orange), and PPA stations are indicated by a plus (green).

In the EA, the upgraded detector design and firmware are frequently undergoing changes in order to improve their performance and reliability. Many of these checks and changes are documented in chapter 4. The EA consists of twelve stations. The details of its configuration can be found in Fig. 4.1.

Following the analysis works presented in chapter 4, as well as [52], a so-called *pre-production array* (PPA) of another 76 SSDs was deployed in 2018 [53]. Due to limitations in the production of the UUBs, these SSDs are all connected to the existing UBs in place of one of the three WCD PMTs. This provides the opportunity to already study and compare SSD and WCD data with higher statistics than is possible with the EA.While the production of UUBs is still on-going, and the PMTs are still being tested, the deployment of the remaining SSDs is already in progress and can also be seen in Fig. 3.8.

3.4 The CDAS Software

The CDAS of the Pierre Auger Observatory mentioned in section 3.2.1 is a set of hardware and applications built to gather and process the data from the Observatory. This includes building SD events and, with the FD data acquisition system, identifying and combining the data for hybrid events.

Since the beginning of AugerPrime, the CDAS software has undergone many changes and revisions to accommodate the electronics upgrade of the SD, and the additional signals from the SSD and the sPMT. These changes mainly include accounting for the additional PMTs and their calibration, as well as processing the higher sampling rate mentioned in section 3.3.2 and the update the SD triggers need to take advantage of it. In addition to the PMT traces, additional monitoring information such as the HV of the PMTs, the MIP histograms, and the UUB temperatures, are directly extracted using CDAS. This additional monitoring data is critical to the analysis of the EA performance described in chapter 4. For further information and more details on the CDAS software see [54].

3.5 The $\overline{\text{Off}}\underline{\text{line}}$ Framework

Most modern astrophysics experiments, especially ones the size of the Pierre Auger Observatory, must deal with thousands of detector responses from different detector types, which each use different variables to describe the quantities measured. To take care of this, and standardize analyses for all detector measurement types for everyone in the Pierre Auger Collaboration, the $\overline{\text{Off}}\underline{\text{line}}$ Framework was conceived [55]. As such, the X_{max} studies presented in this thesis are based on SD simulations that are carried out within the $\overline{\text{Off}}\underline{\text{line}}$ framework. This framework is written in C++ and builds on three core components, which are visualized in Fig. 3.9.

Detector description: It is a structure that summarizes all available data concerning the different detectors. This includes their layout and time dependent functionality, as well as conditions and environment at the Observatory at the time of a measured event. This is used both for real events, and simulated events, providing the ability to describe different detector configurations, which allows for a wide range of analysis applications.

Event data: The event data structure is build to contain every variable which is used by multiple modules or in high-level analyses. This includes a wide range of quantities from detector responses and atmospheric conditions, to shower parameters such as energy or $X_{\max}$. This separation of data storage and data processing enables the unique interchangeability of modules within an analysis sequence. For real data, it starts out with just the detector responses, while the reconstructed event information is subsequently filled in by the various modules. For simulated events, the first module reads in a simulated particle shower and is followed by an event generator which determines the highest level characteristics of the event, such as time and geometry. Detector responses are then simulated and placed in the data structure. After this the event reconstruction procedure is the same as for real data. In the end the event data structure of both real and simulated events is exactly the same except the simulated event contains additional information concerning the simulation parameters used to create the event data.

Algorithms: Every application performed with $\overline{\text{Off}}\underline{\text{line}}$, from analysis to detector simulation, is built on analysis modules. These are sub-programs which carry out specific tasks, interacting only with the detector description and event data structure. This ensures the compatibility of all models with one another. For example, to carry out an analysis one only has to string together the necessary models inside a so-called *module sequence*. Parameters that are adjustable can then be set in an XML file, called the *bootstrap*. Pre-defined sets of module sequences and bootstraps are available to carry out standard detector simulations, reconstructions, and analyses, that can mostly be used without needing alterations by the user.

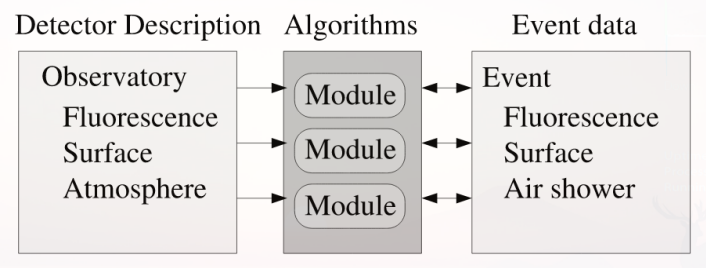


Figure 3.9: Schematic overview of the general structure of the $\overline{\text{Off}}\underline{\text{line}}$ framework. Taken from [55].

3.5.1 Advanced Data Summary Trees

Advanced Data Summary Trees (ADSTs) are a data structure developed within the Pierre Auger Collaboration to contain all high level variables needed for analyses, and optionally low level data [56]. All the necessary information obtained from the above mentioned software components can be stored as an ADST in the ROOT format and be used to browse events and carry out physics analyses. The software to create ADST files can be installed with the Offline Framework or as a standalone.

3.5.2 The Python tool PyIK

The *Python Instrumentation Kit* (PyIK) was developed at the Karlsruher Institut für Technologie. In early versions, it contained a module that was developed to read out variables from ADST ROOT files and to process them directly in Python or save them to files, which are more easily read by Python for later analysis. The module has since evolved into a bigger tool [57] aimed at simplifying common analysis tasks in particle physics, with the ADST reading tool removed to make it available for a broader audience. Hence, an older version of PyIK is used in this thesis to read in the SD simulations for further processing. This version of PyIK is currently not available for download anymore.

4

First Analyses of Engineering Array Data

> *The B-4 is physically identical to me, although his neural pathways are not as advanced. But even if they were, he would not be me.*
>
> — Commander Data, *Star Trek: Nemesis*

Since its deployment in October 2016, the AugerPrime Engineering Array (EA) has continuously taken data. As is the nature of an engineering array, there have been a lot of changes over time. This chapter gives an overview of the general performance of the EA detectors. Several important parameters, such as the detector timing accuracy, the MIP, and the zenith angle efficiency will be studied in this chapter, throughout the various hardware changes. The studies presented in this chapter functioned mainly as an early analysis of the EA data and as a means to gain understanding of the new detector. As such, it is somewhat separate from the remaining content of this thesis. It is, however, referenced in later chapters. Some of the understanding gained here helped to inform the main study of this thesis, such as the parameter choices and evaluating the behavior of the trained network.

4.1 Engineering Array Layout

The EA consists of twelve stations in the configuration seen in Fig. 4.1[1]. The main part of the EA is a hexagon with the regular spacing of 1500 m to understand the expected performance in the full array. However, most of the analyses presented in this chapter will focus only on a twin/triplet set of stations. This is because close proximity both ensures that the statistics needed for an analysis are quickly gathered, and it allows for a range of studies impossible at larger spacing. The triplet is located in the central region of the *SD-433 array*, next to the central station of the hexagon which has the ID 1764 (see Fig. 4.1 and Fig. 4.2). The main characteristic of the SD-433 sub-array of the SD is a reduction in the spacing between WCDs from 1500 m to 433 m. This lowers the energy threshold to 10^{16} eV, and as a result, potentially increases event statistics by six orders of magnitude, as compared to the regular array. Besides those outlined here, further EA analyses are also provided in [52, 53].

[1] Status of April 2019

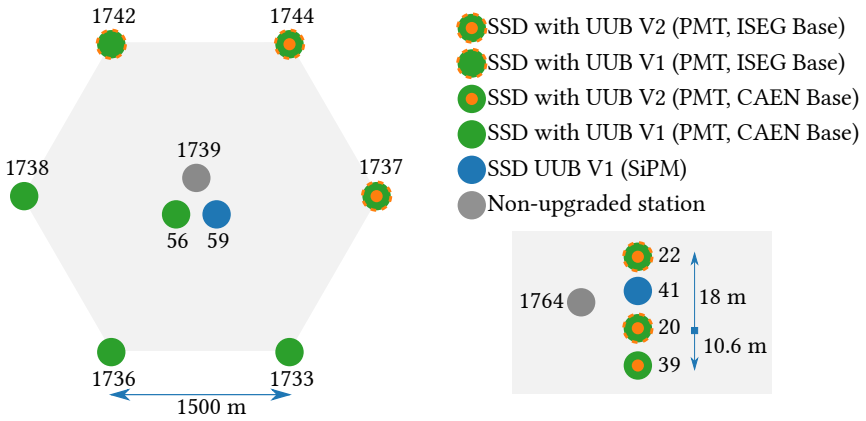


Figure 4.1: Layout of the EA for the upgrade AugerPrime (status 31.03.2019). The main hexagon (left) is part of the regular array with 1500 m spacing. The stations (right) 20, 22, 39, and 41 are in the SD-433 array located elsewhere.

4.2 SD_{433} Reconstruction and HV Stability

The data from the SSDs in the EA has been included in CDAS since October 2016[2]. This includes the four stations located in the SD-433 array, placed next to its central station as seen in Fig. 4.2. Station 41 differs from the other three, as it is taking data with a *silicone photo-multiplier* (SiPM) instead of a regular PMT. Because of this, it is left out of the analyses that are shown in this chapter, as the data cannot easily be compared with that of the other stations. Station 39 is equipped with a regular PMT, however, it was deployed more than a year after the original twin stations 20 and 22. Because of this lag it is not included in the studies, except for the detector timing analysis.

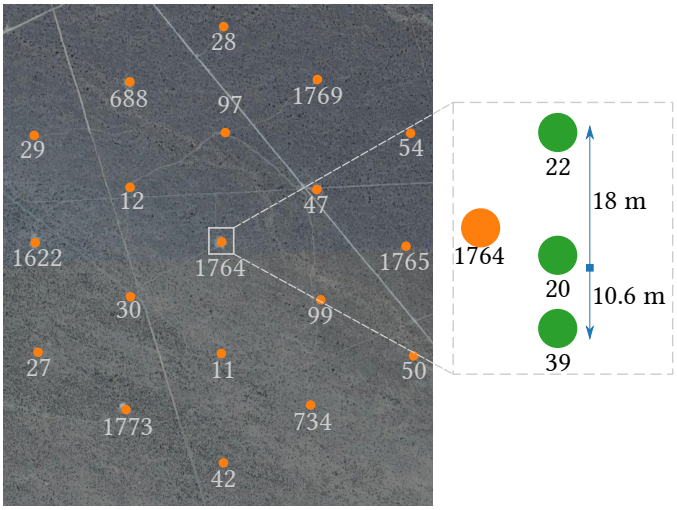


Figure 4.2: SD-433 sub-array of the SD, where the station spacing is reduced to 433 m (status 31.03.2019). Station 1764 is a regular SD station, while 20, 22 and 39 are all upgraded stations.

[2]This data is accessed using ROOT files that contain the raw events

In October 2016, operation of the twin stations began. At this point, the SD-433 array consisted of one full hexagon (seven stations) with station 1764 in its center. The SD-433 array has since been extended (in mid 2017) to include nine additional stations. Two of these are part of the regular spaced array, while four are part of the 750 m sub-array. The remaining three were added only for use in the SD-433 array. The current layout can be seen in Fig. 4.2.

To study the correlation between the SSD data and higher level parameters such as shower direction, shower energy, or distance to shower core, the Offline *Standard Application* `SdInfill433DataReconstruction` (see [58] and [59], hereafter SD_{433} reconstruction) is used to reconstruct the raw event files. Currently the SSD data is read out together with the triggered WCD, but not used in the reconstruction. The twin stations are not included in the SD_{433} reconstruction itself as they are not part of the actual SD-433 array, however, they trigger on nearly all of the SD-433 events as they are next to station 1764. Thus, by pairing the twin station events with the SD-433 array through their Event ID, the SSD data can be combined with the data obtained from the SD_{433} reconstruction.

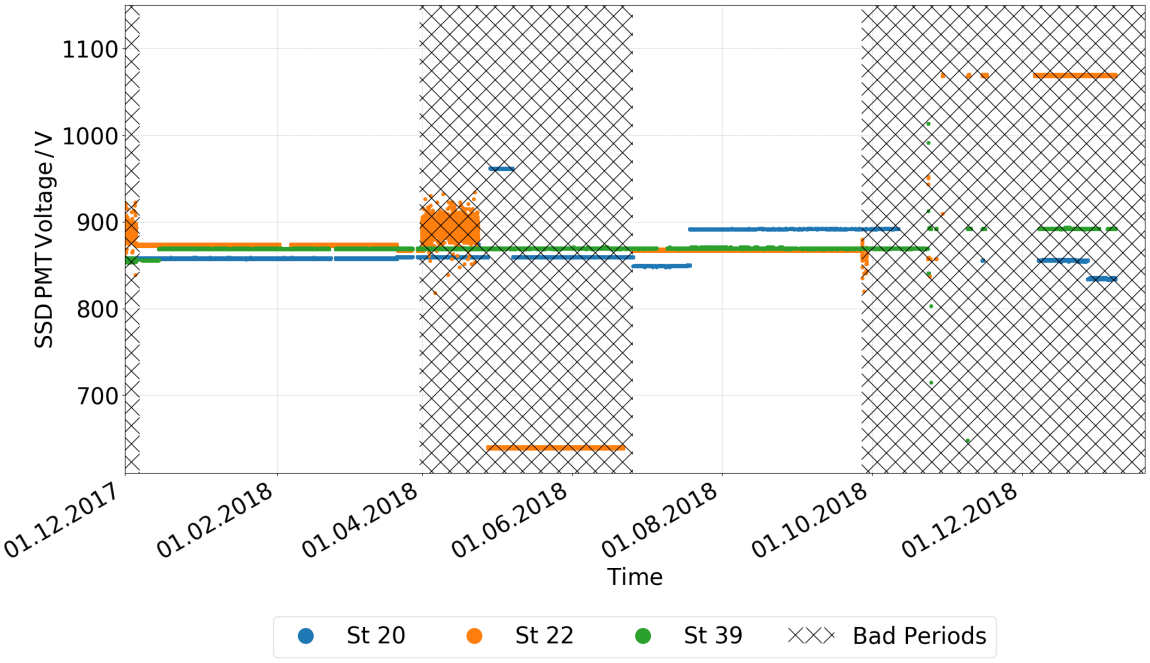


Figure 4.3: Example of the monitored HV of the SSD PMTs over time. Excluded time periods are marked by the shaded regions. Most notably, there are three periods of more 'noise' in the HV for station 22. This behavior of the HV can be seen in the time frames marked as bad periods and originates from the ISEG base used for the PMT used in station 22.

For the comparison of the twin stations the analyzed data-periods are restricted to ensure the stable performance of both stations during the chosen periods. The main determinate of a stable period is the *high voltage* (HV) of the SSD PMTs and so it is constantly monitored to check for changes. Because of this monitoring data it is possible to exclude data where the HV deviates from the default value of about 860–870 V. As

can be seen in Fig. 4.3, an example of the HV settings for the SSD PMT of each station in the triplet, this is very necessary as the HV of each station changed frequently during EA development. For a full overview over the HV for station 20 and 22 only, see Fig. A.1. For the analysis, only data up to 31.03.2019 has been considered. Also, time periods with unstable or deviating HV, as often seen in station 22, are excluded. This leads to only the time periods summarized in Tab. 4.1 remaining for analysis.

Table 4.1: Time periods, where the HV of the SSD PMTs in stations 20, 22, and 39 is stable enough for analysis.

	Date range
HV stable	01.03.2017 - 05.07.2017
	07.12.2017 - 31.03.2018
	25.06.2018 - 26.09.2018
	06.12.2018 - 11.01.2019
	16.01.2019 - 01.02.2019
	05.02.2019 - 31.03.2019

The SD_{433} reconstruction was done for all events measured between 01.03.2017 and 31.03.2019. The distribution of reconstructed shower energies, $E_{SD,rec}$, and zenith angles, $\theta_{SD,rec}$, obtained with the SD_{433} reconstruction are shown in Fig. 4.4. Both of these together illustrate the advantage of a reduced energy threshold in the SD-433 array. Fig. 4.4 also shows that the reconstruction is fully efficient for energies $E_{SD,rec} \geq 10^{16.5}$ eV. The distribution of reconstructed shower cores and the density of events can be seen in Fig. 4.5, as well as the distribution that remains when the reconstruction data is reduced to events that also triggered station 20 and 22.

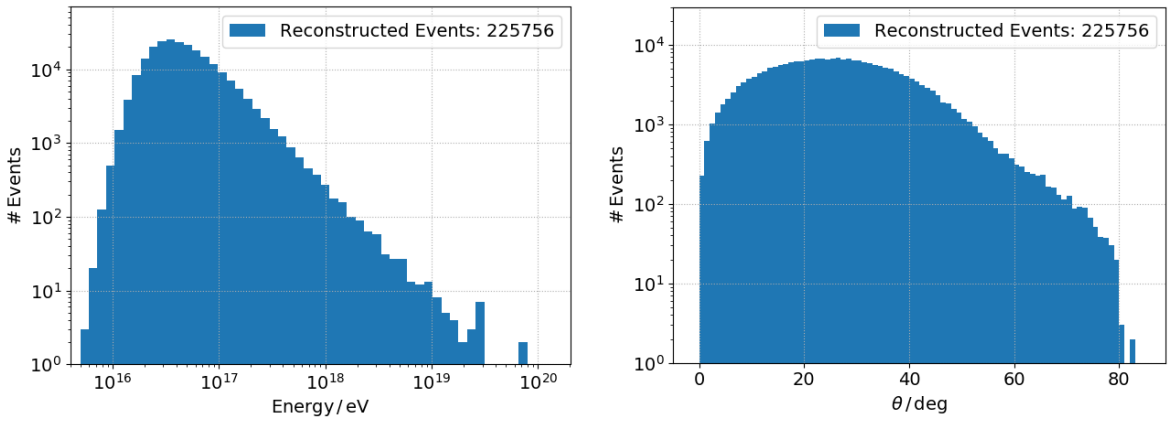


Figure 4.4: Left: Energy distribution of events obtained with the SD_{433} reconstruction. Right: Distribution of reconstructed zenith angles for events obtained with the SD_{433} reconstruction.

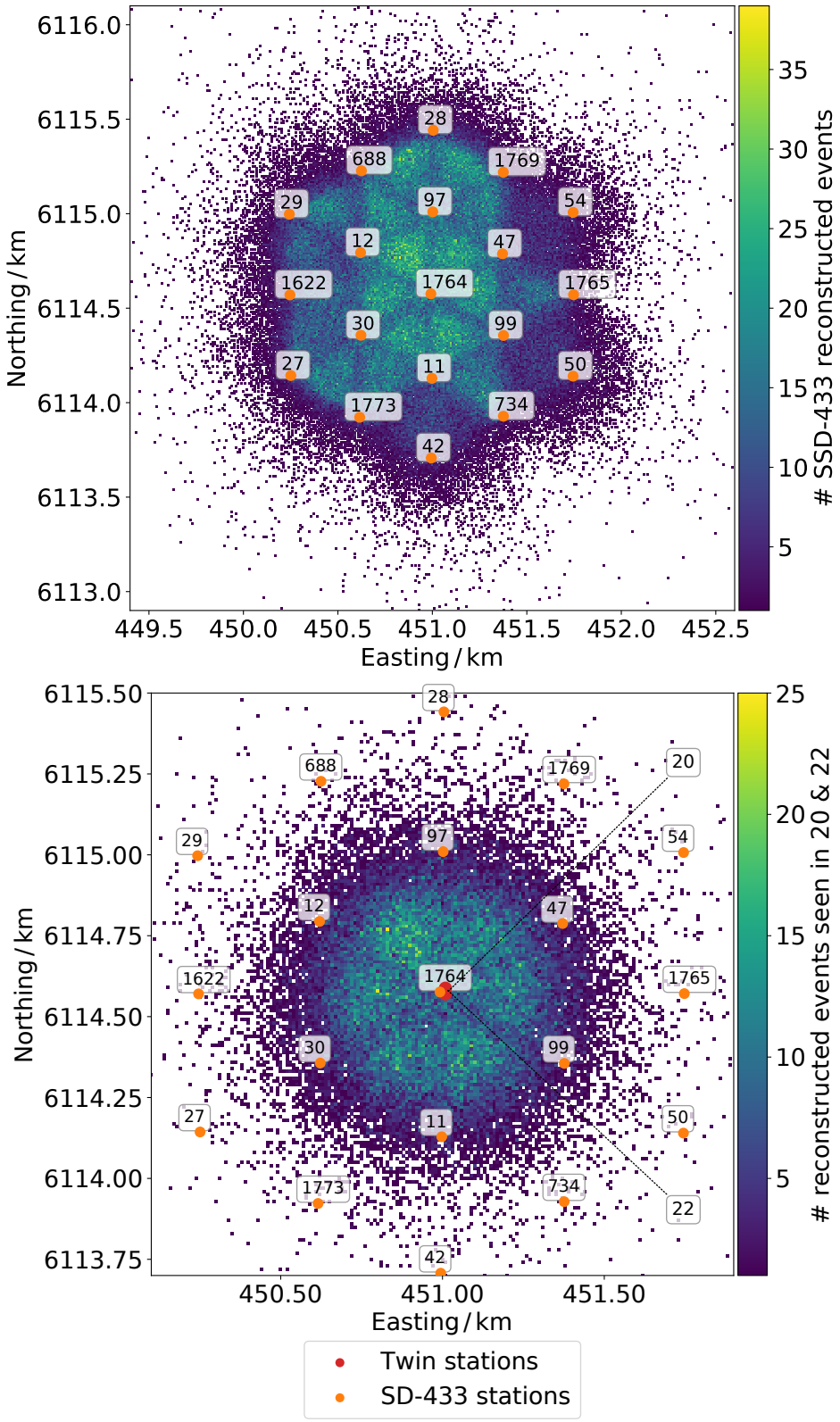


Figure 4.5: Distribution of reconstructed shower core positions obtained with the SD_{433} reconstruction. Top: All SD_{433} reconstructed events. Their density is represented by the color scale. The stations which are part of the SD-433 array are shown as orange dots, together with the station IDs. Bottom: A zoom on the distribution of only the events also detected by stations 20 and 22 (shown in red at the center).

4.3 Charge Estimation of SSD Data and Correlation between Twin Stations

The raw traces of the SSD signals measured with the twin stations are used to calculate the charge of an event. This is done, as at the time this research was carried out, no MIP-calibration was available. Each trace has a sampling rate of 120 MHz, resulting in the upgraded bin width of 8.33 ns. For each SSD PMT, there are two types of traces, the HG and the LG, as mentioned in section 3.3.2. During the studies shown here, the trigger system of the UUBs ran in a 40 MHz compatibility mode. This means the full information registered by the UUB in 120 MHz is down-sampled internally to 40 MHz for compatibility with the standard trigger code used by the rest of the Observatory at this time.

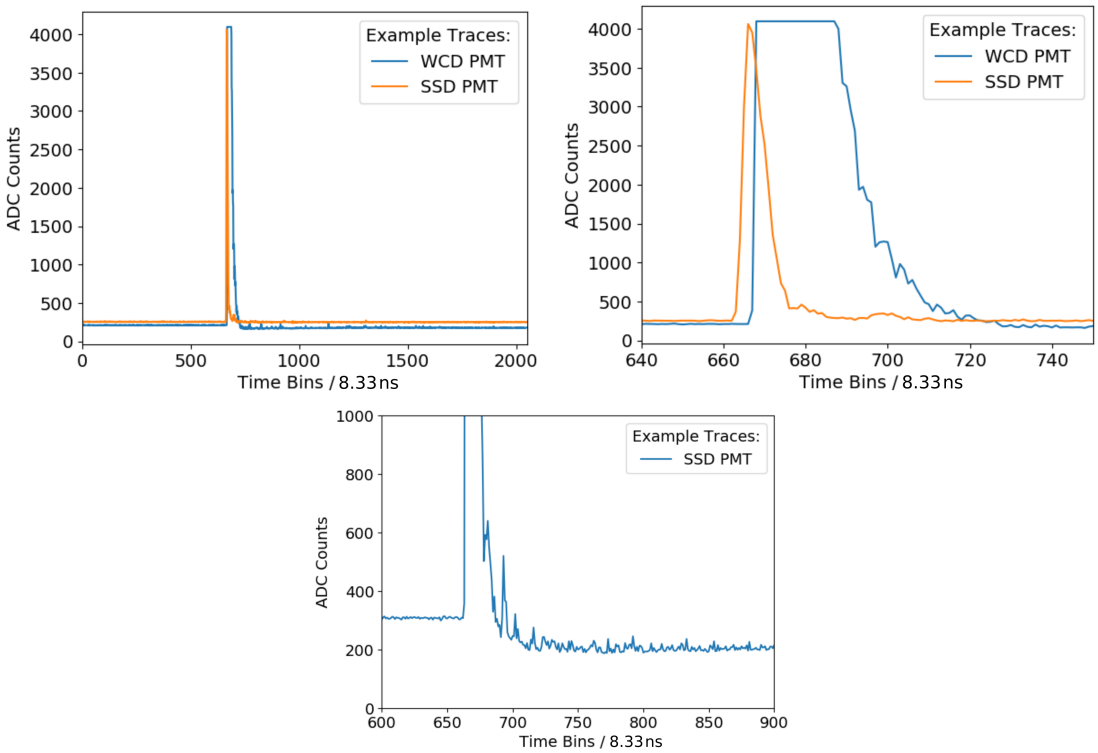


Figure 4.6: Top row: Example HG SSD trace of an upgraded station compared to a WCD trace of the same station. While the WCD PMT is already saturated, the SSD PMT is still below the threshold of 4095 ADC counts and the trace is much sharper. The trace on the right is zoomed in on the important part of the signal. Bottom: Example SSD trace from early data with an undershoot after the main signal peak.

An example of the much narrower SSD trace can be seen in Fig. 4.6 in comparison to a saturated trace from a WCD PMT of the same station. For this analysis, a baseline estimation of each trace is needed and obtained by calculating the mean over the ADC counts in the first 500 bins, which should be signal-free. As can be clearly seen in Fig. 4.6 at the bottom, the traces obtained from the earliest data of the EA display an undershoot after the actual signal. This was caused by a capacitor on the UUB and was fixed by

replacing them in later versions of the UUBs. Because this analysis uses data from both before and after this UUB update, the charge of the traces in this analysis must be calculated in a way which avoids this undershoot in the early data. In practice, this is done by summing up the baseline-subtracted ADC counts of a trace until the signal drops below zero. As a consequence of this method, all signal from late muons is lost, resulting in a slight underestimation of the charge.

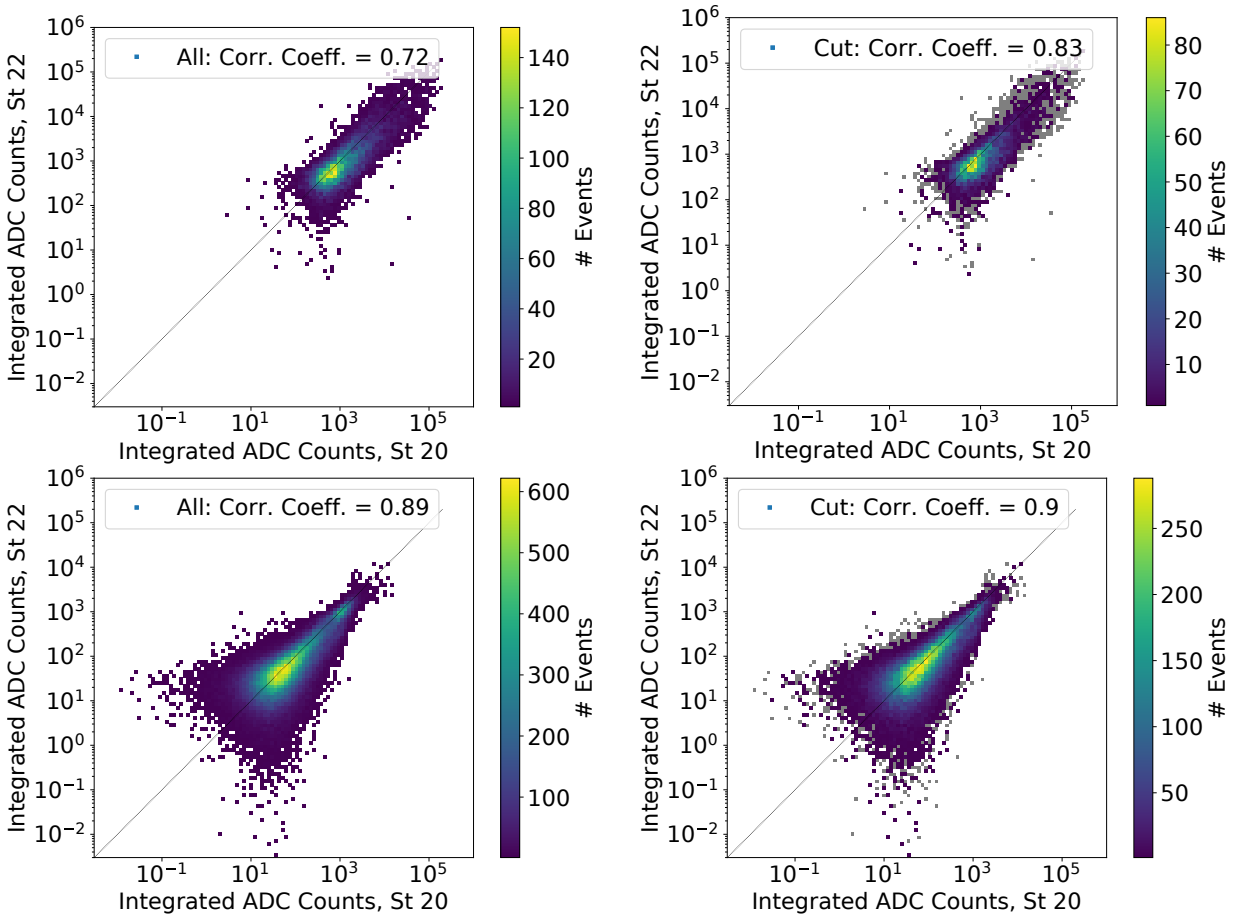


Figure 4.7: Left: Correlation of charges of LG (top) and HG (bottom) events. Right: Correlation of charges of LG (top) and HG (bottom) events, with cut on distance to shower core applied as described in eq. (4.1) (color), compared to uncut results (underlying, in gray). The HG is divided by 32 and the LG multiplied by 4, to make them comparable.

The correlation of the estimated charges between the twin stations for events detected with both stations and contained in the SD_{433} reconstruction is shown in Fig. 4.7. To make the correlations comparable between channels the HG is divided by 32 and the LG multiplied by 4. To improve data quality, events where the maximum value in a trace is below 30 ADC counts or 10 ADC counts, for HG and LG respectively, are discarded. Overall, the correlation between both stations improves with increasing charge, as is expected. It can be nicely seen how the HG and the LG channel line up, with the events with the highest charge in the HG making up for the lower charges in the LG. For the

LG the events with high charge are likely events with a shower core close to the station, which leads to the trace saturating. In this case the charge estimation is not as precise, due to part of the signal missing, which can be a reason for the larger spread.

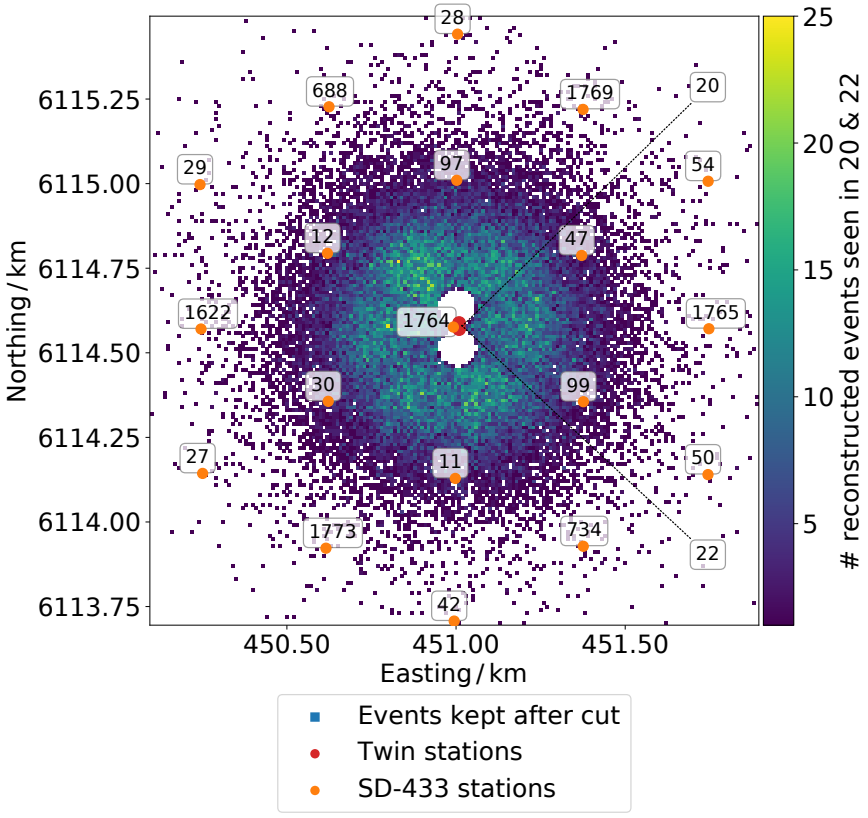


Figure 4.8: Distribution of reconstructed core positions obtained with the SD_{433} reconstruction. The events that were discarded by applying the distance cut are indicated by empty circles around the twin stations 20 and 22. The positions of theses stations are shown as red dots in the center. The color scale shows the density of events.

At this point, no quality-cuts beyond the ADC threshold have been applied to the data. However, a minimum distance cut based on the distance, $R_{20/22}$, between each respective station and the shower core is applied. This cut, which removes direct or close hits to the stations which tend to saturate the traces, should improve the results. To do this, all events meeting the criteria

$$\frac{2|R_{20} - R_{22}|}{|R_{20} + R_{22}|} \geq 0.15 \tag{4.1}$$

are removed. When this cut is applied, the data shows a clear narrowing of the spread of the charge correlation and extends the narrowing to lower signals, especially in the case of the LG channel. The effect the cut has on the charge correlation of the two stations can be seen in Fig. 4.7 on the right. A narrowing of the spread is visible in both HG and LG for high charges and verified by the increase of the correlation coefficients also shown in Fig. 4.7. The remaining positions of the SD_{433} reconstructed shower cores after discarding

the affected events are shown in Fig. 4.8.

4.4 Estimated Charge vs. Distance to Shower Core/Zenith Angle

In this section, $E_{\mathrm{SD,rec}}$ obtained from the SD_{433} reconstruction is used to weight the estimated charges of the SSD signals. This weighted charge is considered vs. distance to reconstructed shower core (see Fig. 4.9), as well as vs. reconstructed zenith angle (see Fig. 4.10). Fig. 4.9 shows, that an advantage of the HG is the ability to detect signals far from the shower core, due to the high amplification of that channel. Conversely, the LG is more suitable for studying events with a shower core close to the station, as the HG traces usually get saturated. These differences become clear when studying the profiles in Fig. 4.9.

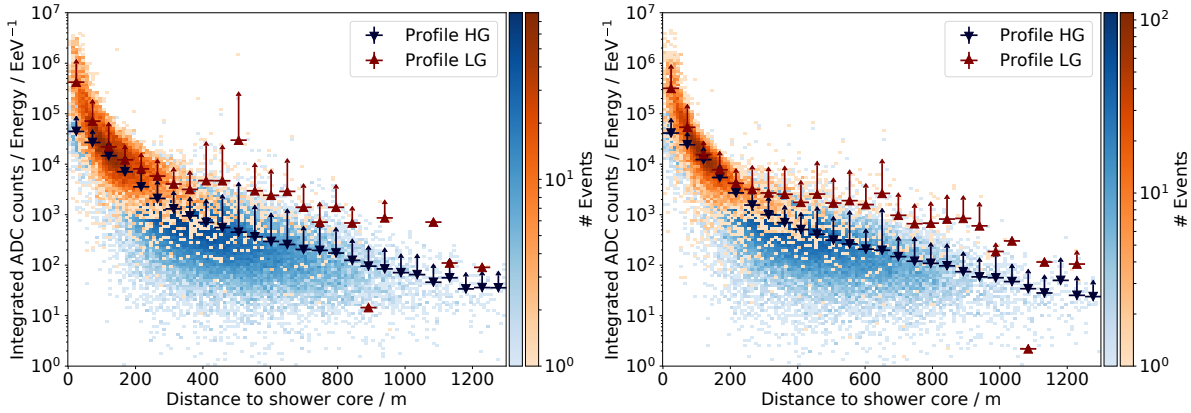


Figure 4.9: Estimated charge weighted with $E_{\mathrm{SD,rec}}$ vs. distance to reconstructed shower core for station 20 (left) and 22 (right). The HG is divided by 32 and the LG multiplied by 4, to make them comparable. The color bars indicate the number of events in a bin.

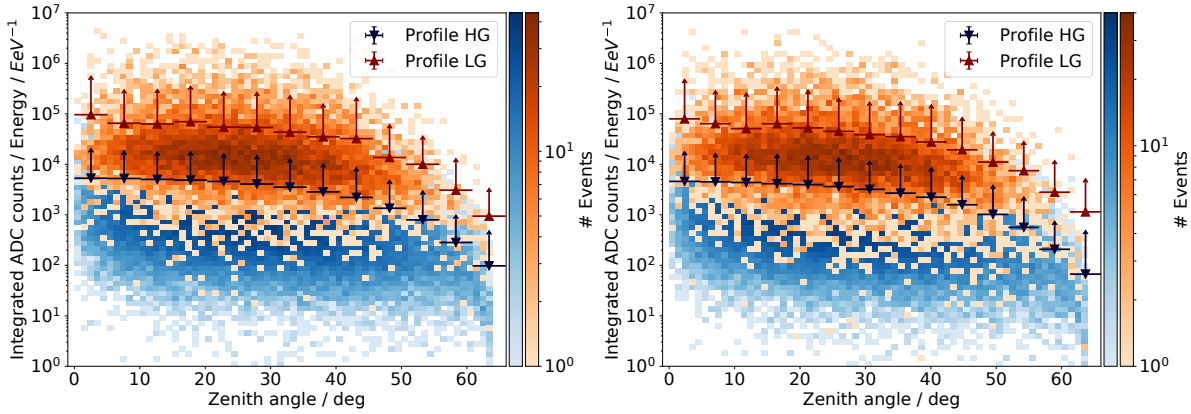


Figure 4.10: Estimated charge weighted with reconstructed shower energy vs. distance to reconstructed zenith angle for station 20 (left) and 22 (right). Same way of representation as in Fig. 4.9.

In Fig. 4.10 the efficiency of the SSD is shown for zenith angles from 0° to 65°, the full zenith range used in normal SD reconstructions. From this, it is evident that in both the LG and the HG, the efficiency of the SSD drops off above $\approx 50°$. This effect is expected given the design of the SSD. Its flattened shape means that its effective area decreases with increasing zenith angles.

4.5 Timing Resolution of the SSD

To get a stronger handle on the time resolution of the SSD and UUB, having a variety of stations with differing spacing is useful. Because of this, the data from station 39 is included when analyzing the resolution of the new detectors. Since station 39 was only placed next to station 20 and 22 at the beginning of December 2017, only data taken after this date can be used. Additionally, like the charge study, the analyzed time periods are restricted to those with a stable HV performance in all three stations. After those two criteria, only two time periods from Tab. 4.1 are used for the timing analysis, which are:

- 7.12.2017 - 31.03.2018
- 25.06.2018 - 26.09.2018.

There are later time periods available, however the timing study was concluded for this thesis after 26.09.2018.

4.5.1 Timing Analysis without Shower Direction Correction

The stations are all equipped with a GPS module to obtain precise timing information [60]. In this analysis, the time information from the GPS module in the form of the GPS nanosecond of the trigger is used to calculate the difference in timing, Δt_{GPS}, between each pairing of the triplet stations using cosmic ray events as the signal source. For comparison with other analyses and to ensure that the events in this study are of good quality, only events that passed the SD_{433} reconstruction are included.

The results of the differences in timing for these events can be seen in Fig. 4.11 and in the appendix as Figs. A.3 and A.5. A Gaussian is fit to the Δt_{GPS} distributions to obtain the spread, σ, of the difference in timing. It can be assumed that both stations contribute equally to the combined resolution, so the timing resolution of a single station can be obtained by dividing this quantity by a factor of $\sqrt{2}$. The results of all possible pairings are listed in Tab. 4.2.

In comparison, the expected timing precision of the GPS modules is ca. 6 ns. In addition, because the stations are being run in the compatibility mode, the sampling frequency of 40 MHz also has to be considered. These two factors add in quadrature to give an expected

resolution of

$$\sigma_{\text{theo}}^{\text{single station}} = \sqrt{\sigma^2_{40\text{MHz sampling rate}} + \sigma^2_{\text{GPS module}}} \tag{4.2}$$

$$= \sqrt{\left(\frac{25\,\text{ns}}{\sqrt{12}}\right)^2 + (6\,\text{ns})^2} \tag{4.3}$$

$$= \sqrt{(7.2\,\text{ns})^2 + (6\,\text{ns})^2} \tag{4.4}$$

$$\approx 9.4\,\text{ns} \tag{4.5}$$

for a single station. The results in Tab. 4.2 show a much larger spread than this and big differences between the different pairings, which scale with the distance between the station pairs. This is easily explained as an effect of the different arrival time a shower front from an EAS has at each station, which depends on the geometry of the shower.

Table 4.2: Timing resolution of the three possible station pairs.

Station pairing	$\sigma_{\Delta t_{\text{GPS}}}^{\text{single station}}$ / ns
20 - 22	21.6 ± 0.4
20 - 39	15.9 ± 0.5
22 - 39	31.1 ± 0.7

4.5.2 Timing Analysis with Shower Direction Correction

The arrival time of the shower front can of course be accounted for. Using the reconstructed zenith angle, $\theta_{\text{SD,rec}}$, and azimuth angle, $\phi_{\text{SD,rec}}$, of the detected shower, and assuming planar shower front[3], the expected timing difference, Δt_{exp}, between any two stations can be calculated as

$$\Delta t_{\text{exp}} = -\frac{\vec{d}\,\vec{n}}{c}\,, \tag{4.6}$$

where $\vec{d}$ is the distance vector between the stations, $\vec{n}$ is the shower axis vector and c is the speed of light. The distance vectors between station pairs are as follows:

$$\vec{d}_{20-22} = (0, 18, 0)^{\text{T}}\,\text{m} \tag{4.7}$$

$$\vec{d}_{20-39} = (0, -10.6, 0)^{\text{T}}\,\text{m} \tag{4.8}$$

$$\vec{d}_{22-39} = (0, 28, 0)^{\text{T}} \tag{4.9}$$

while the shower direction vector is

$$\vec{n} = (\sin(\theta)\cos(\phi), \sin(\theta)\sin(\phi), \cos(\theta))^{\text{T}}\,. \tag{4.10}$$

[3] A curved shower front was also used, but yielded no significant improvement in the timing accuracy, due to the small distances between the stations.

The zenith and azimuth angle dependencies of Δt_{exp} for each event can be seen in Fig. 4.12 (also Figs. A.2 and A.4), along with those of Δt_{GPS}. All distributions of Δt_{GPS} show a similar shape as the corresponding Δt_{exp}. With Δt_{exp} in hand, the shower direction corrected GPS timing is simply

$$\Delta t_{\text{corr}} = \Delta t_{\text{GPS}} - \Delta t_{\text{exp}} \,. \tag{4.11}$$

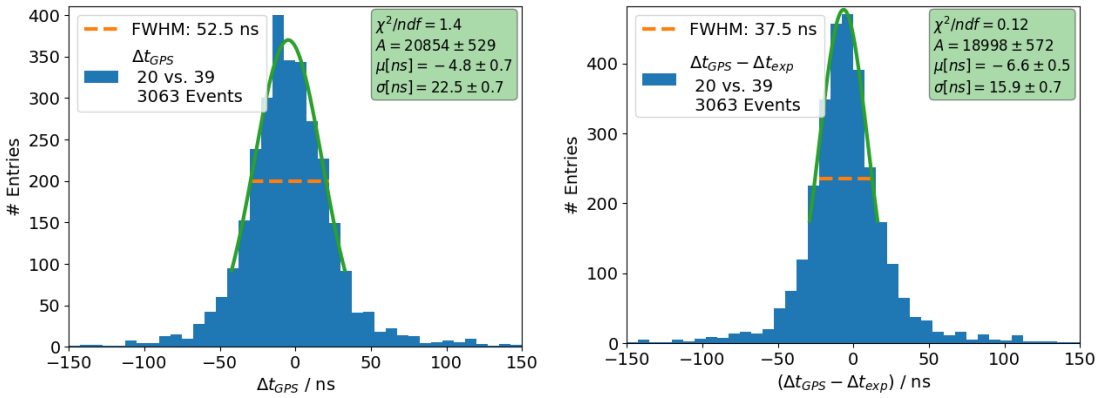


Figure 4.11: Left: Distribution of Δt_{GPS} for stations 20 and 39 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 20 and 39 with Gauss-fit.

The resulting distributions for the corrected timing can be seen in Fig. 4.11 (also Figs. A.3 and A.5). A Gaussian is fit to the distributions of Δt_{corr}. The resulting σ is now an estimate for the combined timing resolution of the GPS devices and sampling rate resolution in the two stations in each pair. In order to obtain the timing resolution the width of the distribution $\Delta t_{\text{corr},20-39}$ can again be divided by a factor of $\sqrt{2}$. The resulting single station resolutions are listed in Tab. 4.3.

Table 4.3: Timing resolution of the three possible station pairs after correcting for Δt_{exp}.

Station pairing	$\sigma^{\text{single station}}_{\Delta t_{\text{corr}}}$ / ns
20 - 22	11.8 ± 0.4
20 - 39	11.2 ± 0.5
22 - 39	12.7 ± 0.7

The results now agree well with each other and are only slightly larger than the expected 9.4 ns given earlier. The reason for this could be shower-to-shower fluctuations, uncertainties in the geometry reconstruction, or simply an overestimation of the GPS resolution.

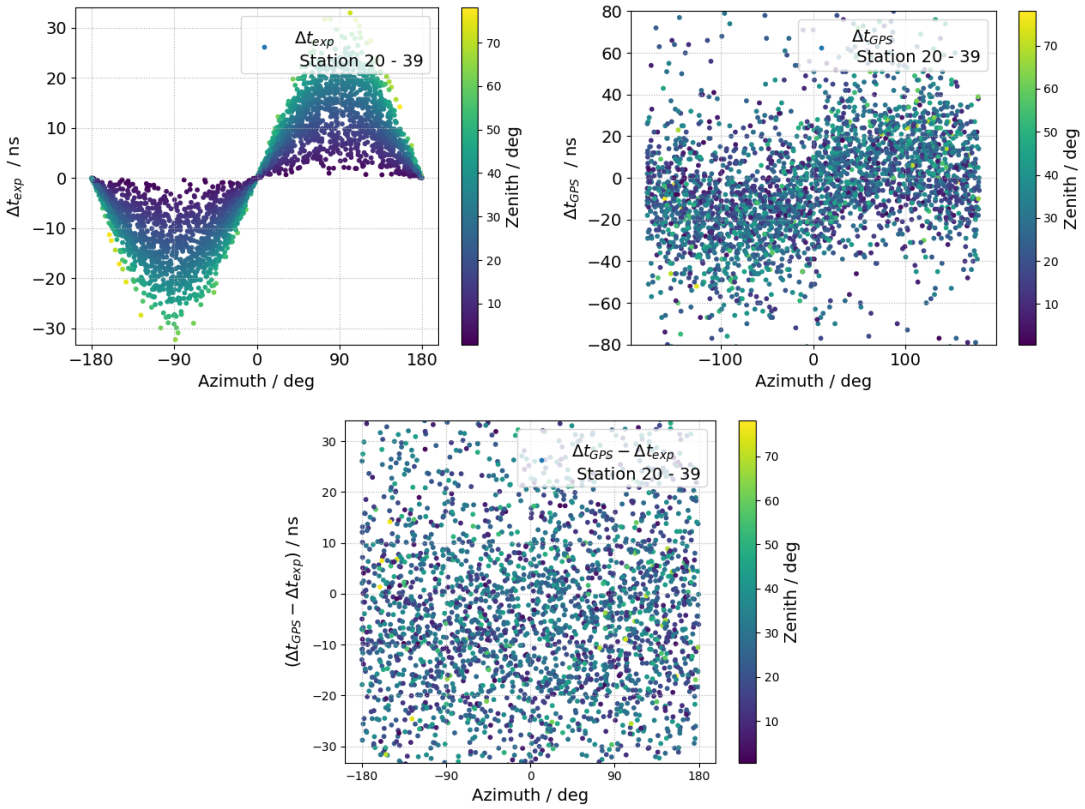


Figure 4.12: Top left: Expected timing differences between station 20 and 39, based on SD_{433} reconstruction depending on azimuth angle of the arriving event. The color scale indicates the zenith angle of each event. As expected for high zenith angles the timing difference increases, and a peak occurs at azimuth angles around $\pm 90°$, as the stations align along this azimuth angle. Top right: Same presentation as on the left, but with timing difference based on observed GPS differences. The behavior is similar to the expected timing distribution, however much more noise is seen. Bottom: The shower direction corrected station timing obtained through the subtraction of the expected timing from the GPS timing.

4.5.3 Timing Analysis during Full-Bandwidth Trigger Period

As mentioned earlier the trigger system of the EA stations runs in the 40 MHz compatibility mode. However, during the short time frame from 22.04.18 to 09.05.18, the triplet stations were run with the *full-bandwidth* (FB) 120 MHz trigger mode to check if this can improve the timing results. The same analysis procedure as before was applied to only the data from this period and the results can be seen in Fig. 4.13 (also Figs. A.6 and A.7). The single station resolution results from fitting the timing distributions with a Gaussian are listed in Tab. 4.4.

The resulting values from comparing stations 20 and 39, as well as 22 and 39 are in general reduced compared to the results with the compatibility mode in section 4.5.2. The result for stations 20 and 22 is much larger. When considering the distributions in Fig. A.6 it is clear that due to the low statistics a Gauss-fit is not feasible.

Table 4.4: Timing resolution of the three possible station pairs after correcting for Δt_{exp} in FB trigger mode.

Station pairing	$\sigma^{\text{single station}}_{\Delta t_{\text{corr}}}$/ns
20 - 22	18.4±13.4
20 - 39	9.9±0.7
22 - 39	9.9±2.1

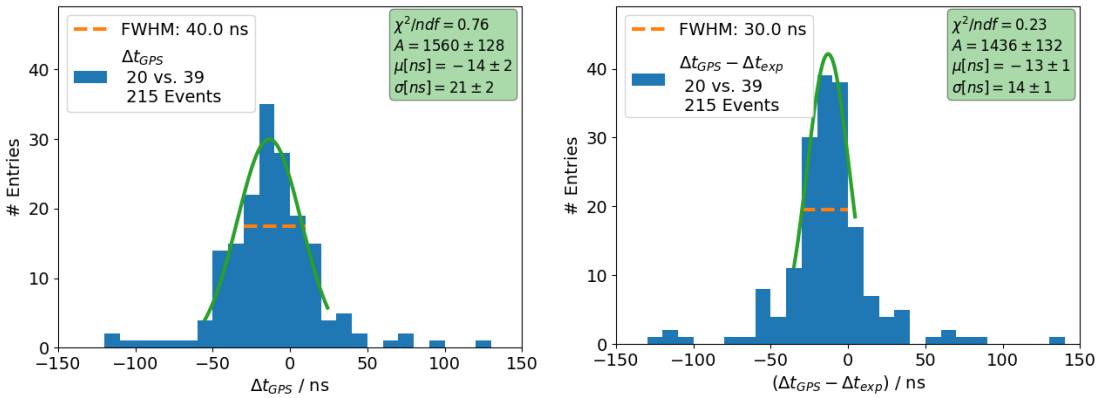


Figure 4.13: Left: Distribution of Δt_{GPS} for stations 20 and 39 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 20 and 39 with Gauss-fit.

A single station resolution as low as 9.9 ns is found, suggesting a good timing resolution improvement which agrees with the theoretical expectation within uncertainties. However, the available statistics for this study are much too low to draw well-founded conclusions. This is partly due to the short time frame of roughly half a month, but can also be attributed to a noticeable drop in the event rate when the FB trigger is enabled. This was likely caused by a problem with the trigger which was written for 40 MHz. Because the trigger threshold window is only a few bins wide the sampling rate leads to a search window which is three times shorter than that of the 40 MHz sampling, which greatly reduces trigger efficiency. This is reflected in the fact that stations 20 and 39, which are the closest, have the highest statistics for this study.

4.5.4 Timing Analysis after switch of GPS module

From 17.07.2018 onward the stations 20 and 39 were equipped with a different GPS module[4]. Additionally, after 25.10.2018 all three triplet stations used a newer revision of the UUB, here referred to as UUB V2. This part of the timing analysis is focused on analyzing the differences in the timing resolution caused by the updated GPS receivers.

The same procedures as in the previous section are used, resulting in the distributions shown in Fig. 4.14 (also Figs. A.8 and A.9). The results show an overall improvement in the precision of the timing after the switch of the GPS receiver with a significant drop in

[4]SSR-6Tf instead of the previously used M12M receivers

Δt_{corr} for the station 20 and 39 pair, which are both using the new GPS module. After this study, the new GPS receiver was chosen for use in the full AugerPrime upgrade. It is also interesting to note that an offset of $\approx 60\,\text{ns}$ can be seen in all timing plots containing data from station 22, which still uses the old GPS module.

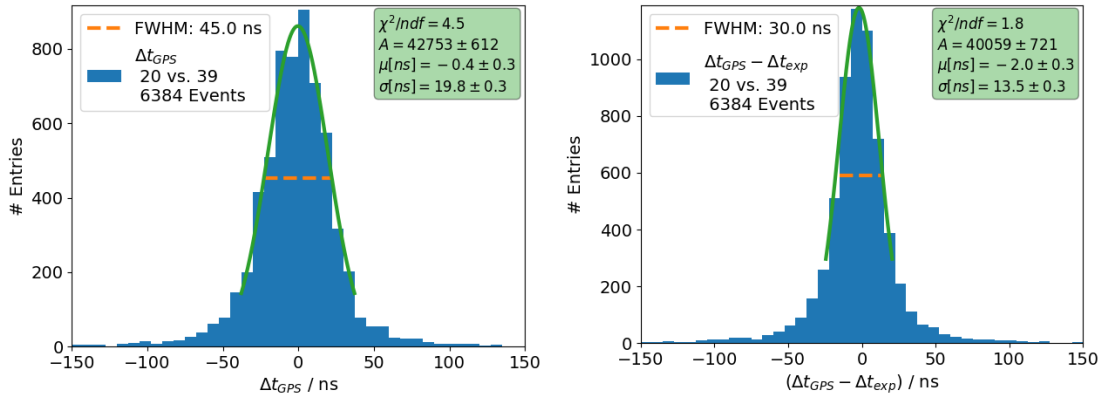


Figure 4.14: Left: Distribution of Δt_{GPS} for stations 20 and 39 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 20 and 39 with Gauss-fit.

Table 4.5: Timing resolution of the three possible station pairs after correcting for Δt_{exp}. Station 20 and 39 are equipped with the new GPS module.

Stations pairing	$\sigma_{\Delta t_{\text{corr}}}^{\text{single station}}$ / ns
20 - 22	12.7 ± 0.5
20 - 39	9.5 ± 0.2
22 - 39	12.0 ± 0.4

The single station resolutions for the respective Δt_{corr} are summarized in Tab. 4.5. The resolution of a single station with upgraded GPS receiver is

$$\sigma_{\Delta t_{\text{corr}}}^{\text{single station}} = \frac{13.5 \pm 0.3}{\sqrt{2}} = (9.5 \pm 0.2)\,\text{ns}\,, \tag{4.12}$$

which is compatible with the expectation of 9.4 ns given in eq. (4.2).

4.5.5 Cut on Signal Size and Analysis of the Signal Start Time

With the expected timing resolution reached, the additional effects of imposing a cut on the minimum SSD signal amplitude is studied for the stations using the new GPS module (20 and 39). The change in $\sigma_{\Delta t_{\text{corr},20-39}}$ depending on the location of the cut can be seen in Fig. 4.15. It is clear, that a cut on the total signal does not provide a significant improvement concerning the spread of the timing distribution. This suggests that the true hardware resolution is being reached.

As a last check, the CDAS implementation for the calculated SSD start bin (the bin in the SSD trace where the actual signal starts) is used instead of the trigger time, to study

if this yields further improvement. The start of a signal is defined in CDAS as the time of the second of two adjacent bins with a signal which is above a certain threshold. For this analysis, a threshold of $3\sigma_{\text{baseline}}$ above baseline was chosen. The difference between the signal start times of the traces for stations 20 and 39, $\Delta t_{\text{startbin}}$, are shown on the left in Fig. 4.16. This difference is then added to Δt_{corr} as

$$\Delta t'_{\text{corr}} = \Delta t_{\text{corr}} + \Delta t_{\text{startbin}} \,. \tag{4.13}$$

The results can be seen on the left in Fig. 4.17.

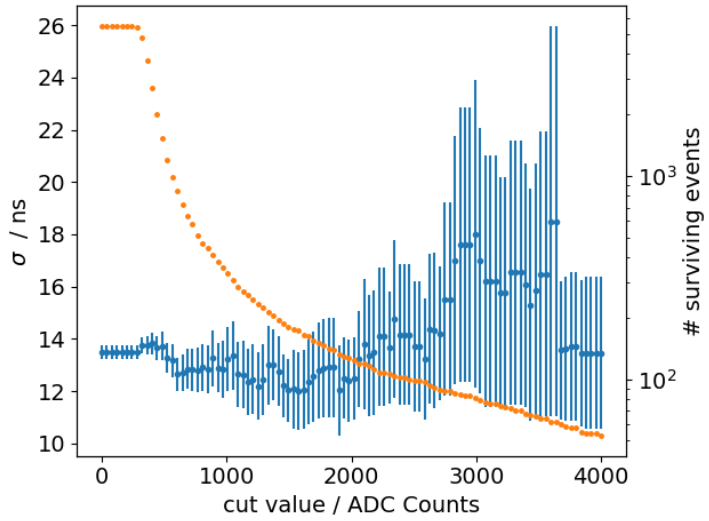


Figure 4.15: Changing of $\sigma_{\Delta t_{\text{corr},20-39}}$ (blue) depending on the strength of the signal cut. Events are kept only if the signal amplitude in both stations is greater than the given cut value. On the second y-axis (right side) the number of surviving events (orange) is plotted to illustrate the drop in event numbers.

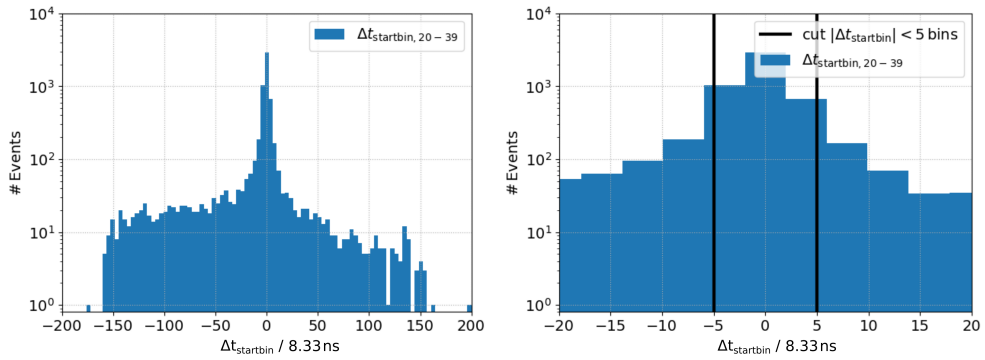


Figure 4.16: Left: Distribution of $\Delta t_{\text{startbin}}$ for stations 20 and 39. A difference of 1 bin equals a time difference of 8.33 ns. Right: Zoomed in on the center region of the distribution around 0, with a cut of $|\Delta t_{\text{startbin}}| < 5$ bins indicated by black lines.

The resulting $\sigma_{\Delta t'_{\text{corr}}}$ is slightly worse than those seen in Fig. 4.14 where the start bin

difference was not included. To try and improve this, the start bin distribution is cut to keep only events with $|\Delta t_{\text{startbin}}| < 5$ bins. The results after this cut can be seen on the right side of Figs. 4.16 and 4.17. Even with this cut the results are not as good as those without the inclusion of $\Delta t_{\text{startbin}}$. This is likely due to a poor identification of the signal start time, and while further studies to improve the calculation of the start bin could be performed, $\sigma^{\text{single station}}_{\Delta t_{\text{corr}}}$ already meets the theoretical expectation value of the hardware. Because of this, more in-depth studies of the detector timing were not pursued.

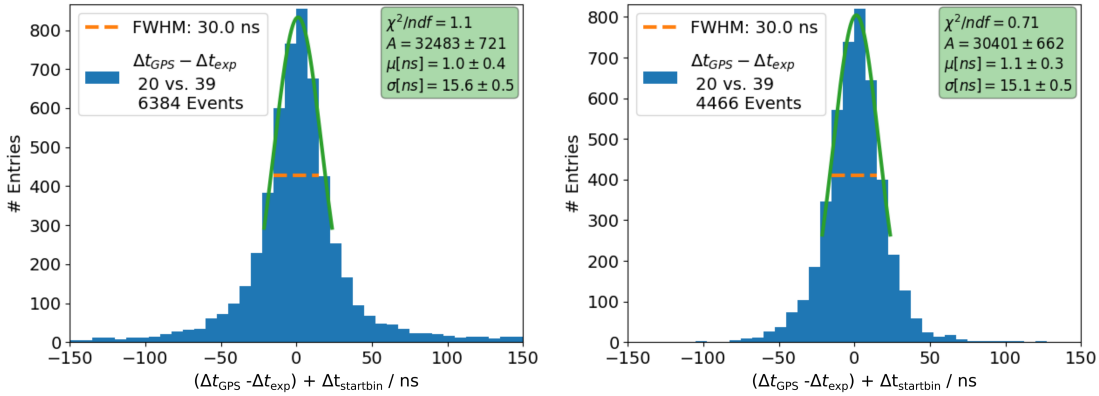


Figure 4.17: Left: Distribution of $\Delta t'_{\text{corr}}$ for station 20 and 39 without the cut on $\Delta t_{\text{startbin}}$. Right: Distribution of $\Delta t'_{\text{corr}}$ for station 20 and 39 with the cut on $\Delta t_{\text{startbin}}$, discarding events where $|\Delta t_{\text{startbin}}|$ differs more than 5 bins, i.e. ≈ 40 ns.

4.6 Annual Variation of SSD PMT Temperature

The temperature of the SSD PMTs is monitored using a calibrated resistor at the PMT base. The temperatures measured by this set up over time can be seen in Fig. 4.18, for stations 20 and 22.

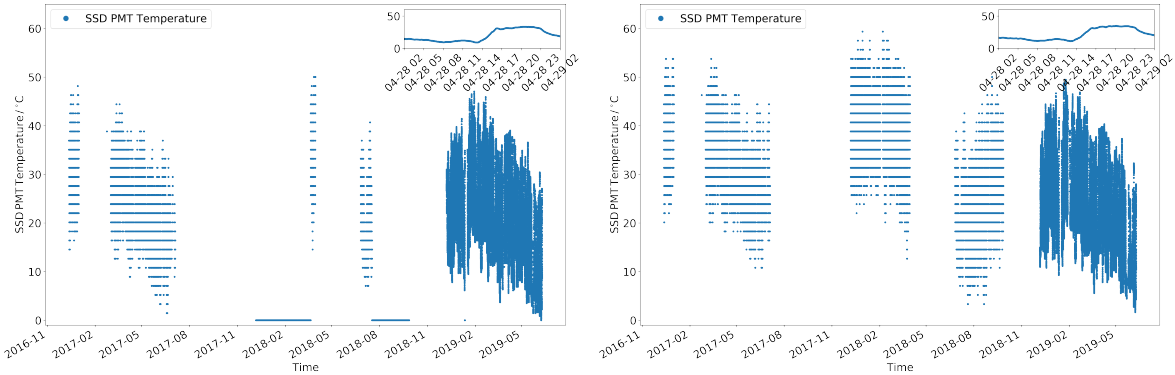


Figure 4.18: SSD Temperature for station 20 (left) and 22 (right) vs. time. A seasonal variance is clearly visible, as well as a daily one (zoomed part). The change in data structure is explained by a change in the read-out.

The expected daily and seasonal variations are clearly seen. However, there is a visible difference in the PMT temperatures starting in November 2018, which coincides with the switch of the UUB in both stations from version 1 to version 2. This difference is explained by changes in the readout of the data with CDAS. The PMT temperature is reported using 16 bits, however, only 12 bits are necessary to represent the expected temperature range. However, in the beginning, only 8 of the 12 bits were read out. This error was caught, resulting in the much finer resolution seen after November 2018.

4.7 MIP Peak Stability vs. Time/Temperature

As mentioned in section 3.3, the most crucial parameter for the signal calibration of the SSDs is the MIP. It is calculated, using the location of the peak in the charge histogram, an example of which can be seen in Fig. 4.19 together with the calculated MIP peak. A typical charge histogram consists of two main features, the noise peak and the MIP peak. In this thesis, to calculate the MIP peak, first the running average of the charge histogram is calculated in order to smooth the data. The maximum points of the resulting smoothed curve are then calculated using the bin to bin difference quotient, as the point where its value switches from a positive to a negative value corresponds to the maxima of the smoothed charge histogram. Ideally the charge histogram should only have two maxima, first the noise peak, followed by the MIP peak, but occasionally multiple peaks are observed due to the prototype state of the detectors and read-out. Even so, a clear temporal evolution of the calculated MIP peak for both station 20 and 22 can be seen in Fig. 4.20 as the blue data points. The HV of the corresponding SSD PMT is also depicted (in orange), to show possible correlations.

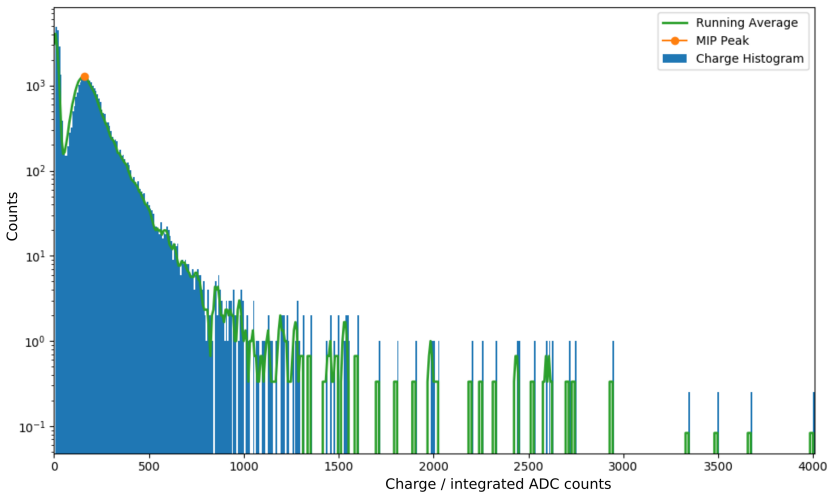


Figure 4.19: Example of MIP peak calculation from a charge histogram of station 22.

The temperature of the SSD PMT is also correlated with the MIP, using only the stable HV periods, and shown in Fig. 4.21 and Fig. A.10, for stations 20 and 22 respectively.

To quantify the temperature stability of the MIP, a line is fit to the data, and the results are summarized in Tab. 4.6. For station 20 the data is split into two populations after the switch to UUB V2, which is due to small differences in the HV. Overall, the results vary between stations and UUB revisions. For both UUB versions, considering a daily change in temperature of around 20 °C the MIP peak varies at maximum by around 20 integrated ADC Counts. As can be seen from the two different distributions for station 20 with UUB V2, it is clear that the MIP peak is also sensitive to changes in the HV. The effective signal changes $\frac{\Delta\text{MIP-Peak}}{\Delta T}$ (where ΔMIP-Peak is the relative change in %) are also shown in Tab. 4.6. Results between the two stations and UUB revisions are comparable considering the sensitivity to slightly different HVs, except for UUB V1 of station 22 where the effective signal change is much lower. Both of these findings could be investigated in more detailed studies, but further analyses did not fit the time frame of this thesis.

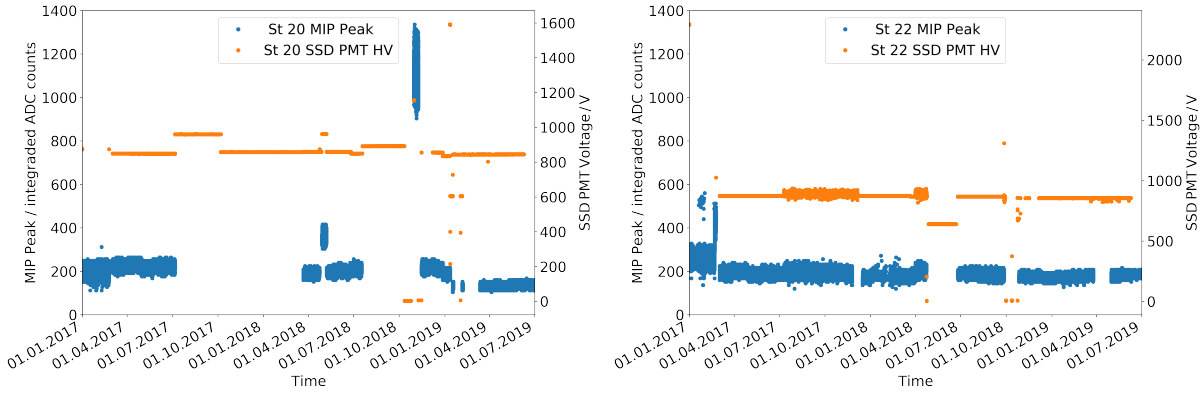


Figure 4.20: MIP peaks calculated for station 20 (left) and 22 (right) vs. time. The HV of the SSD PMTs is shown in orange. The instability of the HV in station 22 has a minor effect on the spread of the MIP peak position. Overall the MIP peak is more unstable in station 20 than in station 22.

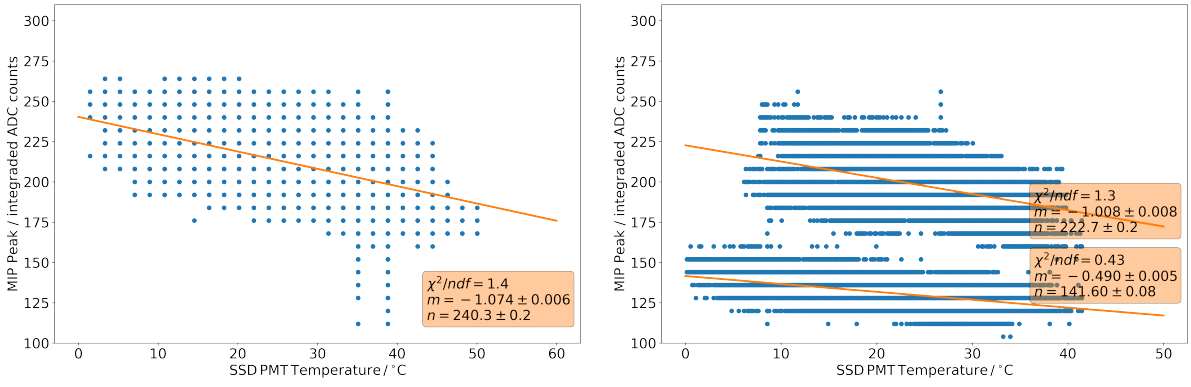


Figure 4.21: MIP-temperature correlations for station 20 with UUB V1 (left) and UUB V2 (right). The refined temperature calculation is visible on the right. The two distributions on the left are due to different HV settings.

Table 4.6: Summary of fit values for MIP-temperature correlations in station 20 and 22 for both UUB revisions.

Stations	Time frames	m / $\frac{\text{int. ADC Counts}}{°\text{C}}$	n / int. ADC Counts	$\frac{\Delta\text{MIP}-\text{Peak}}{\Delta T}$ / $\frac{\%}{°\text{C}}$
20	UUB V1	-1.074 ± 0.006	240.30 ± 0.20	-0.447 ± 0.003
	UUB V2 a	-1.008 ± 0.008	222.70 ± 0.20	-0.453 ± 0.004
	UUB V2 b	-0.490 ± 0.005	141.60 ± 0.08	-0.346 ± 0.004
22	UUB V1	-0.036 ± 0.004	189.20 ± 0.10	-0.019 ± 0.002
	UUB V2	-0.853 ± 0.003	194.61 ± 0.08	-0.438 ± 0.002

4.8 Trace Baseline Stability

As mentioned in section 4.3 the baseline of the SSD PMT trace is taken to be the mean over the first 500 bins of the trace. The stability of the baseline over time can be seen for stations 20 and 22 in Fig. 4.22 and Fig. A.11, respectively. The dependence of the baseline on the temperature of the PMT is shown in Fig. 4.23 and Fig. A.12. Due to the change in temperature read-out mentioned in section 4.6, the temperature dependence is evaluated separately for UUB V1 and V2. There are several distinct populations of baseline-temperature correlations visible, each indicated by a different color scale to identify the time period of the data. The time dependence of the temperature dependence is clearly reflected in Fig. 4.22 and Fig. A.11 by the jumps in the baseline.

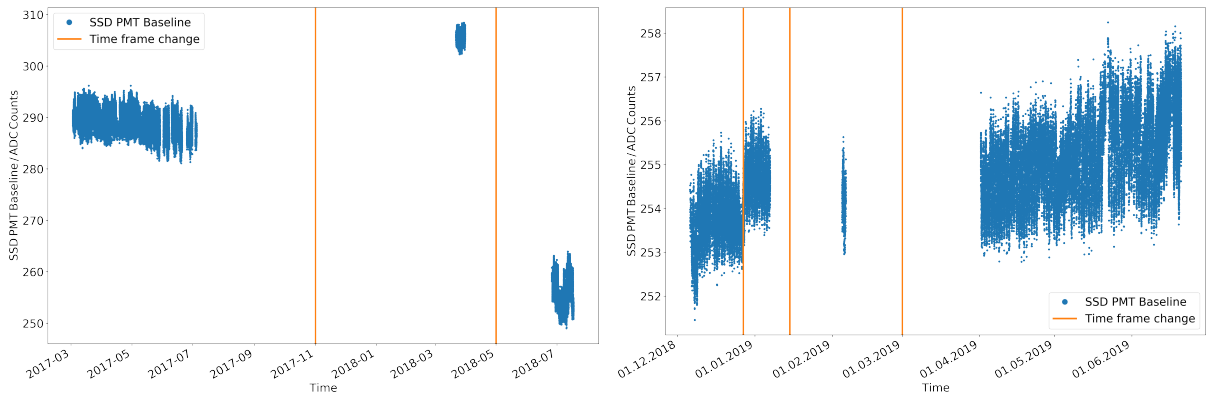


Figure 4.22: Baseline vs. time for station 20 with UUB V1 (left) and UUB V2 (right).

To parameterize the temperature dependence and its changes over time, the data is grouped into different time periods, so that the temperature correlation of each period can be separately fit with a line. For data taken with station the division into time periods is shown for both UUB versions in Fig. 4.22, while the distinctive time frames for data station 22 are shown in Fig. A.11. The fitted slopes are also shown in Fig. 4.23 and Fig. A.12 with the resulting fit values displayed in boxes next to the respective line-fits. The results of this procedure are summarized in Tab. 4.7.

From this analysis, it is clear that the switch from UUB V1 to V2 decreases the temperature dependence sharply, as indicated by the much lower slopes of the line fits. Ad-

ditionally, the individual slopes for each period of the UUB V2 data agree well within uncertainties. The fact that the sign of the UUB V2 slopes differ between station 20 and 22 can likely be attributed to their small values and minor differences in the station components. Considering again a daily change in temperature of around 20 °C, this results in a baseline variation of less than 1 ADC Count, which is negligible. This means, that with UUB V2 the baseline of the SSD traces is very stable with temperature.

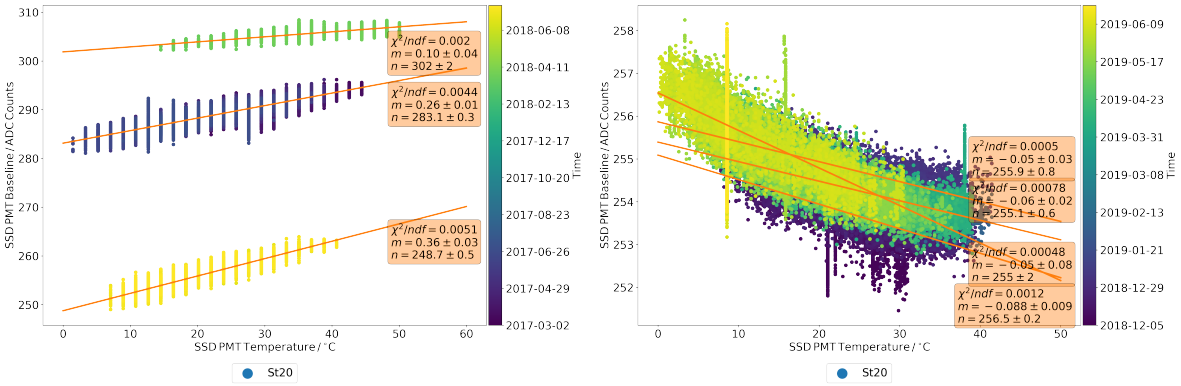


Figure 4.23: Baseline-temperature correlations for station 20 with UUB V1 (left) and UUB V2 (right).

Table 4.7: Summary of fit values for baseline-temperature correlations in station 20 and 22, differentiated by time interval and UUB version

Stations	Time frames	$m / \frac{\text{ADC Counts}}{°\text{C}}$	n/ADCCounts
20	UUB V1 a	0.26 ±0.01	283.1 ±0.3
	UUB V1 b	0.10 ±0.04	302 ±2
	UUB V1 c	0.36 ±0.03	248.7 ±0.5
	UUB V2 a	−0.06 ±0.02	255.1 ±0.6
	UUB V2 b	−0.05 ±0.03	255.9 ±0.8
	UUB V2 c	−0.05 ±0.08	255 ±2
	UUB V2 d	−0.09 ±0.01	256.5 ±0.2
22	UUB V1 a	0.18 ±0.01	296.8 ±0.3
	UUB V1 b	0.46 ±0.04	252 ±1
	UUB V1 c	0.43 ±0.03	248.4 ±0.6
	UUB V2 a	0.04 ±0.02	251.6 ±0.6
	UUB V2 b	0.05 ±0.03	249.7 ±0.9
	UUB V2 c	0.03 ±0.02	250.3 ±0.4
	UUB V2 d	0.02 ±0.02	248.7 ±0.4
	UUB V2 e	0.03 ±0.06	249.9 ±0.7
	UUB V2 f	0.01 ±0.05	251.1 ±0.7
	UUB V2 g	0.01 ±0.04	253.5 ±0.4

4.9 Conclusions from the EA Analysis

The various studies presented in this chapter have shown definite improvements in the data quality over time for the EA, and it is clear that the prototype stations of the EA show a promising performance of AugerPrime. The resolution of the detector timing matches well with what is expected from hardware. Additionally, both the MIP peak and the baseline of the SSD traces have gained stability with improvements in the UUB. Further development was put into the components of the UUB, which is, at the time of this thesis, in its final revision. An understanding of the details of this new detector was gained and the knowledge can be transferred to the composition studies with AugerPrime simulations that are the subject of the following chapters.

5
Artificial Neural Networks & Deep Learning

> *"Computers make excellent and efficient servants, but I have no wish to serve under them."*
>
> — Mr. Spock, *Star Trek: The Original Series*

Machine Learning is the building of any kind of model using samples of data referred to as *training data*, that is capable of making predictions or decisions, using computer algorithms. Due to their often complex and multivariate problems, physics challenges are well suited to the application of machine learning. At this point, most modern physics experiments use some form of machine learning to evaluate their data. This is especially true in particle physics, with their often large amounts of data, real and simulated, and computational resources available.

5.1 Basic Components of a Neural Network

Artificial neural networks (ANNs), also known simply as *neural networks* (NNs), are a common type of machine learning algorithm which has been used since the 1960s [61]. Interestingly, they were developed to model the biological neural functions of human brains. Like a brain, to accomplish a task well a NN must learn first. To do this, an NN is presented with example sets of data, the training data, which contains the input information of the problem and also the solution in the form of an already known output. The NN then learns how to connect these two, and after having processed enough training data, eventually gains the ability to predict the output based only on the input information.

A basic schematic of a NN can be seen in Fig. 5.1, which illustrates the main components of a NN. At its base, every NN consists of *neurons* or *nodes*. Fundamentally, these contain information the network uses to learn. These nodes are grouped in *layers* where they operate together at different *depths* in the network. Each NN consists of an input layer, whose nodes represent the input variables, and an output layer, which represents the variable(s) the network aims to predict. Every other layer between the input and output layers are called a *hidden layer*, which contain *hidden nodes*. These hidden layers are where the network learns the different aspects of the data needed to form its predictions. A simple NN has at least one hidden layer, but the terms *deep learning* and *deep neural network* (DNN) arise when an NN has more hidden layers, an example of which can be seen in Fig. 5.1. The hidden nodes in a layer are connected with the nodes from both the

preceding and the following layers via *weights*. They are used to determine which nodes pass on their information to the next layer and to what extent. To calculate the content of any hidden node, y_j^l, its incoming weights, w_{ij}^l, are multiplied with the values from its corresponding previous nodes, y_i^{l-1}, and summed as

$$y_j^l = f\left(b_j^l + \sum_{i=1}^{n} w_{ij}^l \cdot y_i^{l-1}\right), \tag{5.1}$$

where b_j^l is and optional additional bias. Here, w_{ij}^l is called the weight matrix, and b_j^l is called the bias vector, of the current layer l. These are combined with the node content vector y_i^{l-1} from the previous layer $l-1$. The end output of this process is then evaluated with an *activation function*, f, which, as the name suggests, determines how strong, if at all, a node is activated. As a last note on the terminology, if every node in a layer is connected to every node in the next layer, the network is *fully connected*.

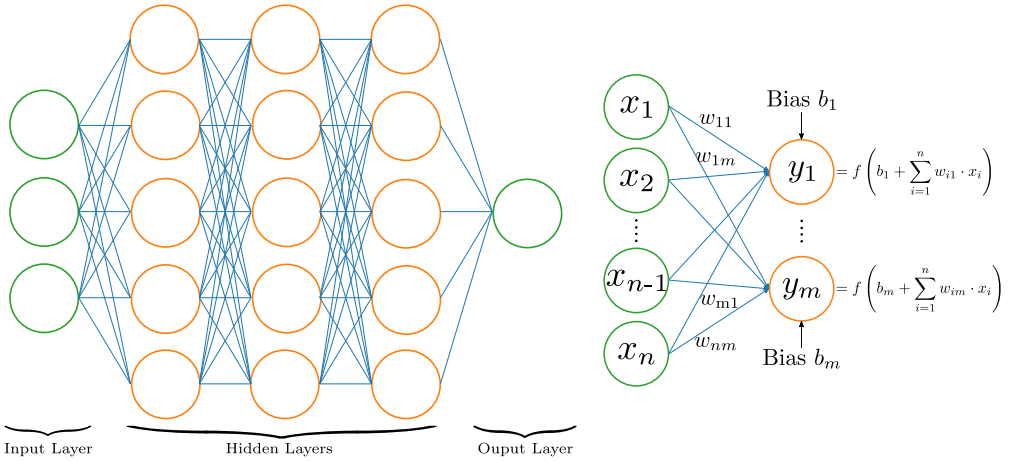


Figure 5.1: Left: Schematic view of a fully connected DNN. The green circles represent the nodes in the input and output layers.The orange circles show the nodes in the hidden layers. The blue lines connecting the nodes represent the weights. Right: Application of weights and bias to the values $y_i^{(l-1)=0} = x_i$ of the input layer to form the contents $y_j^{l=1} = y_j$ of the current layer.

Various different types of activation functions exist and are usually grouped by whether they are *linear* or *non-linear*. Linear activation functions will not be discussed here as they are unsuitable for the complex task of predicting primary mass:

1. A linear activation function effectively collapses all the hidden layers and the output layer into a single layer, since the network in this case would be a linear combination of linear functions, which is a linear function itself.

2. Linear activation functions make it impossible to use *back propagation* ()see section 5.2.4), as the derivative of the function is a constant and therefore preserves none of the information of its input.

Because of this, most modern NN, and all DNN, use non-linear activation functions, as these do allow both for back propagation and having multiple hidden layers. Of these, the most commonly used are the *Sigmoid*, *TanH*, and *ReLU* functions pictured in Fig. 5.2.

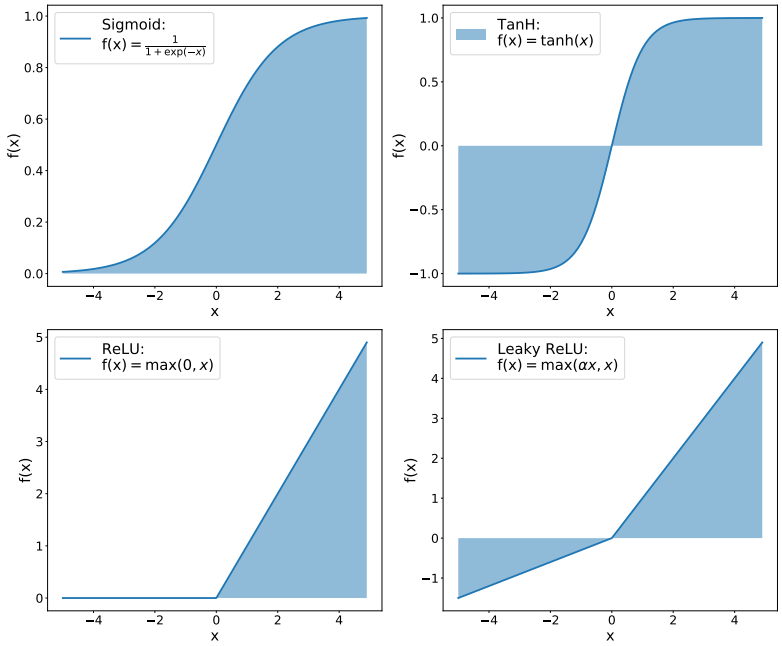


Figure 5.2: The four most common non-linear activation functions.

Sigmoid: The Sigmoid function is defined as

$$f(x) = \frac{1}{1+\exp(-x)} \quad \in (0,1)\,. \tag{5.2}$$

It is useful as it normalizes the output of a neuron, since every value is mapped to one between 0 and 1. It is often used as the activation function of the output layer to map the predicted values to probabilities, which lends itself to applications in classification problems.

TanH: The TanH function can be written as

$$f(x) = \tanh(x) = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)} \quad \in (-1,1)\,. \tag{5.3}$$

Compared to Sigmoid, the TanH function is zero centered, which lends itself well to problems that have strong negative, neutral and positive input values.

Rectified Linear Unit (ReLU): The ReLU function is of the form

$$f(x) = \max(0,x) \quad \in (0,+\infty) \tag{5.4}$$

and is compared to the previous two much more computationally efficient due to quick convergence. For this reason, it is a good initial choice when training a model. However, this activation function has a disadvantage which is known as the "dying ReLU" problem. As can be seen from its derivative, when an input value is negative the gradient becomes zero which makes back propagation impossible and means the network cannot learn. In order to avoid this issue, an adaption known as *Leaky ReLU* (see also Fig. 5.2) can be used

$$f(x) = \max(\alpha x, x) \quad \in (-\infty, +\infty),\ \alpha \in (0, 1)\,, \tag{5.5}$$

where α is like a scaling factor for negative x that is used to make gradient calculation possible.

5.2 Initialization and Training of a Neural Network

The training process of NNs mainly consists of three steps which will be presented below.

1. Forward propagation
2. Cost calculation
3. Back propagation

The goal of the training process is for the network to learn the best set of biases b_j and weights w_ij for predicting the output. This learning process takes place over a set of training iterations called *epochs*. During each of the epochs, a sequence of the above mentioned three steps is performed.

5.2.1 Data Set Splitting

Before training of the network can take place, the data that is to be analyzed needs split into three sets, the *training*, *validation*, and *test* sets. The training set is a sub-sample of the full data set that is used for the network to train on and learn from. Because of the threat of a network learning the individual events, it usually consists of the biggest part of the full data set. The validation set is an unbiased sub-sample of the full data set and is used to evaluate the model on data that it has not been trained which helps with tuning the NN setup. Last is the test set. The test set is also an unbiased sub-sample of the full data set, but is only used to evaluate the networks performance once it is fully trained. As such, it is also the data set that is used to compare competing models. The decision on what split ratio to use to form these sub-samples, and especially the decision whether or not a validation set is needed, depends on the number of *hyperparameters* (i.e. the weights

and biases) the model needs to fit. With fewer hyperparameters the need a network has for a validation set becomes lower.

5.2.2 Global Parameters, Metric, and Weight/Bias Initialization

In addition to splitting the data set, there are a large number of parameters and settings that have to be chosen. Unfortunately, often they can only be changed and improved after evaluating an earlier trained model making the process of improving a network somewhat tedious. The three most important global parameters are the number of *epochs*, the *learning rate*, and the *batch size*. The number of epochs defines how long the network will train for. The learning rate is a factor used to scale the strength of learning that is applied each epoch of training. A high learning rate increases training speed, but at the cost of often poor convergence behavior. A low learning rate, however, can lead to a network, which trains and learns too slow or not at all. To solve this, there are advanced methods which combat these issues, by e.g. lowering the learning rate when the network stops learning to increase sensitivity. The last global parameter is the batch size, which defines how many samples from the training or validation set the network sees at once. It is also used to estimate the error gradient, which will be discussed below.

To evaluate how well a model performs the *metric* is used. The metric can be calculated after each epoch during training or can be used to compare fully trained networks trying to predict the same result. The choice of the metric is done by the user and is entirely dependent on what the goal of the analysis is.

As mentioned before the NN tries to find the optimal set of biases and weights to make a specific prediction from a set of inputs. However, at the beginning of the training process the weights and biases are not necessarily all set to 1 but, but instead can be initialized in a number of ways. The most common initialization is a random sampling from a normal or uniform distribution for the weights, while the bias is typically initialized as 0. To do this one draws values from a distribution with mean= 0.0 and some finite variance. The goal is to set the variance for the weights so that the variance of the output is the same as the variance of the input

$$Var(y_j^l) = Var(y_i^{l-1})\,. \tag{5.6}$$

In practice, it is a common approach to calculate the variance for the weights with the help of the average number of nodes as

$$n_{\text{avg}} = \frac{n_{\text{in}} + n_{\text{out}}}{2} \tag{5.7}$$

$$Var(w_{ij}^l) = \frac{1}{n_{\text{avg}}} = \frac{2}{n_{\text{in}} + n_{\text{out}}}\,, \tag{5.8}$$

The initialization scheme in eq. (5.7) was developed in [62], is known as *Xavier/Glorot*, and is best used for classification tasks. The best initializer for tasks that use ReLU activation functions was developed in [63], and is known as *He/HeNormal*. He/HeNormal introduces an additional factor of 2 in the variance as the ReLU function cannot have a mean of 0.0

$$Var(w_{ij}) = \frac{2}{n_{\text{avg}}} \,. \tag{5.9}$$

This is the most used initializer for regression tasks.

5.2.3 Forward Propagation and Cost Calculation

The first step in an epoch is forward propagation. This is simply the process in where the training data is given to the network as input, is passed through the layers, applying the activation function at the nodes, until the output layer and prediction are reached.

In the second step, the *cost function* is evaluated on these predicted outputs which compares them to the true output values. Generally, this cost function is tied to the *loss function*, which is a term that is often used synonymously. However, the loss function is used to compute the error for a single training example, while the cost function is the average value of the results of the loss function for every event in the entire training set. The cost is calculated after every epoch and plays a vital part in the learning algorithm of an NN. This is because it is used during back propagation, where the weights and biases are adjusted in order to try and minimize the cost. The cost function is also calculated on the validation set to help evaluate the performance of the model during training. There are different cost functions that can be applied depending on the problem at hand. Broadly they can be sorted into two categories, cost functions for classification problems and cost functions for regression tasks. Generally speaking classification problems deal with predicting discrete values or classes, whereas regression tasks aim to predict continuous values. The goal of this thesis is to make predictions on the depth of shower maximum, X_{max}, of cosmic rays, which is a continuous value, i.e. a regression task.

The cost function, $J(\theta)$, also known as the *objective*, can generally be written as

$$J(\theta) = f(y_j^{\text{true}}, y_j^{\text{pred}}(x_i, \theta)) \tag{5.10}$$

where $\theta = \{w_{ij}, b_i\}$ is the set of the weights and biases matrices, x_i is the network input, and f is the underlying functional form chosen to relate them. There are three variants of cost functions that are well suited for dealing with regression tasks, all of which use the error of the prediction. The most common cost function, and usually the first approach

tried, is the *mean squared error* (MSE). It is defined as

$$\text{MSE} = \frac{1}{N}\sum_{j=1}^{N}\left(y_j^{\text{true}} - y_j^{\text{pred}}\right)^2 \tag{5.11}$$

and works best when the target values follow a Gaussian distribution and are all of a similar magnitude. Since a larger discrepancy between true and predicted value results in a larger error, the larger the mistakes made by the model, the larger the penalization.

A variant of the MSE, used when the target values are distributed over a larger range, i.e. several orders of magnitude, is the *mean squared logarithmic error* (MSLE) which is defined as:

$$\text{MSLE} = \frac{1}{N}\sum_{j=1}^{N}\left(\ln(y_j^{\text{true}}+1) - \ln(y_j^{\text{pred}}+1)\right)^2 . \tag{5.12}$$

The introduction of the logarithm to the equation means that the cost now only focuses on the relative difference between true and predicted value. As a consequence small differences between small true and predicted values are treated similarly to larger differences between large true and predicted values. An example of this behavior can be seen in Tab. 5.1. Additionally, underestimates are penalized more than overestimates, i.e. the error is not symmetric, see Tab. 5.1.

The third variant is the *mean absolute error* (MAE), which can be written as

$$\text{MAE} = \frac{1}{N}\sum_{j=1}^{N}\left|y_j^{\text{true}} - y_j^{\text{pred}}\right| \tag{5.13}$$

and works particularly well, when the target values are mostly Gaussian distributed on a small scale, with large outliers. This is because the algorithm is more robust to these than MSE, as can be seen in Tab. 5.1.

Table 5.1: Example calculations to compare different cost functions. MSE penalizes large errors much more than both MSLE and MAE. Both MSE and MAE penalize the same for over- and underestimation.

	y^{true}	y^{pred}	MSE	MSLE	MAE
Small diff. & numbers	10	20	100	0.4181	10
Large diff. & numbers	10000	20000	100000000	0.4803	10000
Underestimate	20	10	100	0.4181	10
Overestimate	20	30	100	0.1517	10

5.2.4 Back Propagation and Network Evaluation

In the last step of an epoch, back propagation is carried out which functions as the "learning part" of training an NN. During back propagation, the gradient $\nabla_\theta J$ of the

cost function is used to feed the data backwards through the network and thus optimize the weights, biases, and optionally the learning rate or other parameters. The scheme for carrying out this optimization is known as the *optimizer* and is chosen during the initialization phase.

There are various types of optimizers, however, even the more sophisticated ones are commonly based on the very simple method of *gradient descent*, where the cost function is minimized by going in the opposite direction of the gradient $\nabla_\theta J$ at the current point, as this is the direction of steepest descent and leads to a local minimum. The weight and bias matrices (θ) are then updated accordingly for the next epoch ($\hat{\theta}$), by applying the learning rate α as

$$\hat{\theta} = \theta - \alpha \nabla_\theta J(\theta). \tag{5.14}$$

The most simple optimizer calculates the gradients for the whole data set. To perform a single bias/weight update, which makes it slow to converge. Also because it must evaluate the full data set at one, it is unusable in a case where the whole data set does not fit into memory.

An alternative to simple gradient descent is *batch gradient descent*. Here, updates to the weights and biases are applied in batches according to a set batch size. A further adaption of this idea is the *stochastic gradient descent* (SGD), which performs an update for every sample in the data set

$$\hat{\theta} = \theta - \alpha \nabla_\theta J\left(y_j^{\text{true}}, y_j^{\text{pred}}(x_i, \theta)\right) . \tag{5.15}$$

This greatly increases convergence speed, but it does so at the cost of precision.

All three of the above described gradient descent algorithms are only able to adapt the weights and biases. There are, however, other more sophisticated optimizers that not only adapt weights and biases, but also adjust the learning rate. One such example is *Adam* (Adaptive Moment Estimation) [64]. Adam is particularly useful when dealing with noisy gradients. It adapts the hyper-parameters, θ_t, based on the exponentially decaying average of past and past squared gradients as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{5.16}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 . \tag{5.17}$$

Here, $g_t = \nabla_\theta J(\theta_t)$, is the gradient at the current time step, while m_{t-1} is the past gradient and v_{t-1} the past squared gradient. These are updated to m_t and v_t with the help of the hyper-parameters β_1, $\beta_2 \in [0, 1)$, which control the exponential decay rates, and also represent estimates of the first and second moment of the gradients. In the first time step, $m_{t=0}$ and $v_{t=0}$ are initialized as zero vectors which leads to the moment estimates also

being biased around 0. This initialization bias needs to be counteracted, which leads to the following bias-corrected estimates

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{5.18}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t}\,, \tag{5.19}$$

which are then used in the Adam update rule as follows

$$\theta_{t+1} = \theta_t - \frac{\alpha \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}\,. \tag{5.20}$$

In eq. (5.20), ϵ is a smoothing term to avoid divisions by zero. For the hyperparameters, the authors of [64] recommend default values of $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$.

With the completion of this last step, one epoch of NN training is finished. Once this sequence of steps has been repeated for the desired number of epochs, the network is considered trained. Usually a trained network is referred to as a *model*. The model can then be evaluated. Model evaluation is a very problem specific task and the method developed in this thesis will be described in chapter 7.

5.2.5 Combating Under- and Overtraining

During the training of an NN, one of the issues that can arise is *under-* or *overtraining*. In both cases this means the model is not able to predict previously unseen data well. In the case of undertraining this means the model is not even able make a good prediction using the training set. For overtraining the opposite is true and the model has learned the training set so well that is focuses on the specific data, but not the underlying trend, and thus can not predict either the validation or test set [65]. While undertraining can usually be combated by increasing the capacity of the network, the most common reason for overtraining is over-parameterization. Over-parameterization is a phenomenon where the network architecture is too complex for the model to generalize, which results in the above mentioned behavior.

In Fig. 5.3 the training and validation cost curves are shown. These are two of the most obvious identifiers of under and overtraining. Ideally the validation cost should fluctuate around the training cost, but overall follow the same trend. However, sometimes the behavior shown in the aforementioned figure occurs, where after a certain number of epochs a gap forms between the two curves. This separation is known as the *generalization gap*, and is a strong indicator that the performance of the network is not the same for the training and the validation set. This is not necessarily a problem as long as the width of the gap converges against a constant value like the purple dash-dotted line in Fig. 5.3. Once this gap increases in size however, and especially if the curve for the validation loss

starts to trend upwards as is the case for the orange curve in Fig. 5.3, the network is overtraining, and tailoring the network too strongly to the training data. This in the end degrades the performance of the model, making it unable to properly predict results from data it has not seen before.

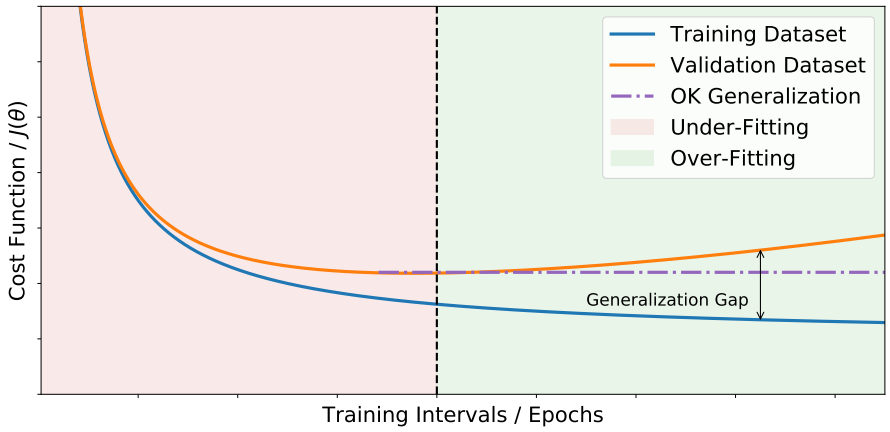


Figure 5.3: Schematic representation of cost for both the training and the validation set vs. training steps. The training cost should generally decrease. If the validation cost is higher than the training cost, the difference is called the generalization gap or error. A common sign of overtraining is if the validation cost has a minimum. A common way of avoiding the latter, is early stopping,i.e. ending the training of the network at the dashed line before the network goes into overtraining.

One of the simplest ways of combating overtraining is to identify the point in the training process where the network transitions into overtraining, which is marked by the vertical dashed line in Fig. 5.3 and then use the trained network from that epoch as the end model. This process is called *early-stopping*. Other options are:

- to make the network smaller by reducing nodes or layers, resulting in less hyper-parameters to fit
- to increase the amount of training data if possible
- to add random noise to the data throughout training, for example by adding random non-functioning stations.

A more sophisticated approach to reduce overtraining is to apply a method called *dropout* [66]. The key idea behind this method is to randomly force nodes into a deactivated state in each layer, resulting in a thinned-out version of the initial network architecture. This process is performed for each batch that is fed into the network. In the simplest implementation, nodes are retained according to a probability p (in [66] a default value of $p = 0.5$ is recommended). The nodes that are kept in the network are then up-scaled by a factor of $1/(1-p)$, in order for the sum over all nodes to be on average unchanged. An

example of such a thinned NN can be seen in Fig. 5.4. During testing, when the model is used to make predictions on as of yet unseen data, the full version of the network is used. The weights that were calculated during training are then down-scaled by the dropout rate, i.e. if the dropout probability in training is p, the outgoing weight of a unit in the model is $p \cdot w$.

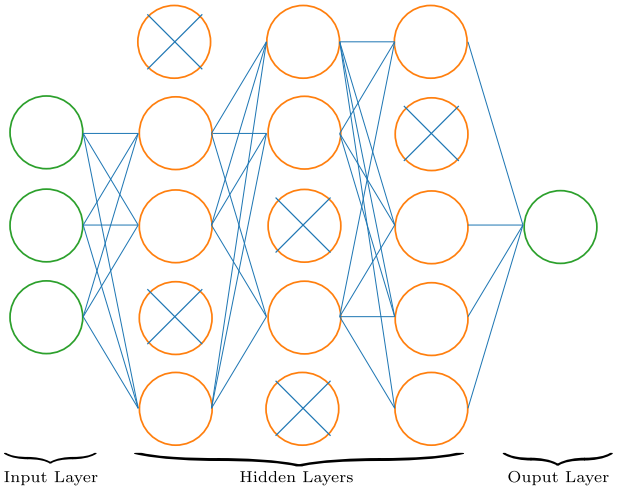


Figure 5.4: Example of a thinned neural network when using dropout. The same architecture as in Fig. 5.1 is used, but some nodes are removed (here crossed-out) and their incoming and outgoing weights set to 0.

Another sign of overtraining in an NN is the size of weights. If some of the weights are very large it can be an indicator of the network being to complex resulting in the training data being over-fit. A method to avoid this is to penalize the network during training based on the size of weights. Typically this is done by applying the L_1 norm $\left|\left|w_{ij}^l\right|\right|_1$ or the L_2 norm $\left|\left|w_{ij}^l\right|\right|_2$ to regularize and encourage smaller weights. This is carried out by adding the w_{ij}^l, when calculating the cost function as

$$J(\theta)_{L_1} = f(y_j^{true}, y_j^{pred}(x_i, \theta)) + \lambda \sum_{i=1}^{n} \left|\left|w_{ij}^l\right|\right|_1 \tag{5.21}$$

$$J(\theta)_{L_2} = f(y_j^{true}, y_j^{pred}(x_i, \theta)) + \lambda \sum_{i=1}^{n} \left|\left|w_{ij}^l\right|\right|_2^2 \tag{5.22}$$

where λ is a scaling factor that defines the overall strength of the penalization [1]. These are then called L_1 or *lasso regularization* and L_2 or *ridge regularization* [65]. In practice, often a combination of the two regularization methods is used, this is called *elastic net regularization*.

[1] It has to be noted that in the case of the L_2 norm, it is actually the squared L_2 norm

5.3 Convolutional Neural Networks

While fully connected NNs can be used for almost any analysis problem, the required number of nodes can quickly become prohibitively expensive. Consider a grid of inputs, $m \times m$, where the information needed to understand the data from any particular grid point is localized to the neighboring grid points. A fully connected NN would produce a sparse net of nodes as the relationship between distant grid points has no predictive power. Instead, a convolution can be applied to the input grid, keeping it locally dense, while removing the zero weight distant nodes from the NN, lowering the dimensionality of the input data.

NN which employ this approach are called *convolutional neural networks* (CNNs). The convolution is carried out by applying so-called *filters*, which are essentially arrays of weights, to the input data and convolving the two in order to identify features. This is done by sliding the filter over the input, with the part of the input currently having the filter applied to it being called the *receptive field*. To do this the depth of the filter has to be the same as the depth of the input, for example

$$(f_{\text{width}}, f_{\text{height}}, Color - Channel_{\text{depth}}) \tag{5.23}$$

in the case of images. During this process the values in the receptive field are multiplied with the contents of the filter and summed up, as shown in Fig. 5.5. The sums for each receptive field/filter pair, are then stored as the output of the convolution in what is called a *feature map*. The size of the feature map depends on how often the filter fits into the input data. Per layer, multiple filters can be used to detect multiple features.

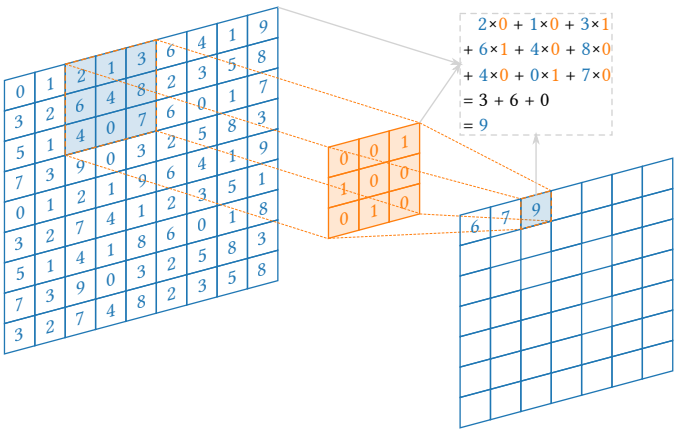


Figure 5.5: Example of a convolutional filter, in orange, being applied to a two dimensional input of size 9×9, with the receptive field in highlighted in blue. Due to the filter size of 3×3, the output will be of size 7×7.

In the first convolutional layer, highly localized correlated features, i.e. straight or curved lines in the case of image pixels, are detected. In the deeper layers of the network, wider correlated features are picked out, with the complexity of the features increasing with the

depth of the network. For example, semicircles (combination of a curve and straight edge) or squares are picked out in the second layer and body parts or full objects are identified in the last layers. The last layer in a CNN is usually a fully connected layer, which combines the data in the last feature map (containing all the previously mentioned features). This layer then makes a prediction.

Another aspect considered in convolutions are the channels an input can have. The most approachable example of this are the color channels in an image (2D) or in every frame of a movie (3D). More generally the channels can be seen as different information carriers for the same respective input. The way convolutions are applied to different channels is visualized in Fig. 5.6, using a 2D color image as an example. As can be seen, the filters used for the convolution are applied to each channel separately and the results are added up to one combined output, to be stored in the feature map. Different types of convolutional layers exist which can be used to create an NN architecture and the choice of layers is highly dependent on the geometry of the task as well as the shape of the input data.

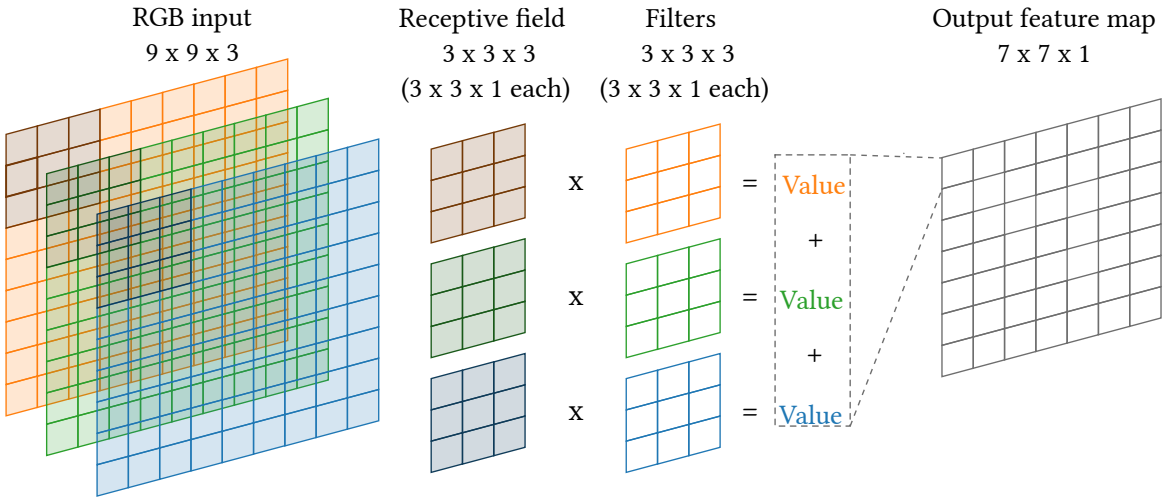
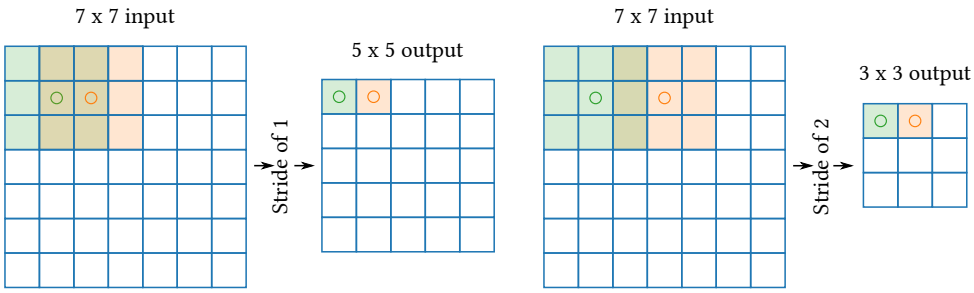


Figure 5.6: Example of convolutional filters being applied to the different color channels. The receptive field of $3 \times 3 \times 3$ is split into three color channels (darker colors) and convoluted with a filter of size $3 \times 3 \times 1$ each. The results of these convolutions are then combined in the feature map.

5.3.1 Striding and Padding

There are several types of settings that can affect how a convolution is performed, which can be set for each individual layer. The two most common parameters are the *striding* and the *padding*. The stride parameter sets the number of steps taken each time a filter is moved across the input and can be set for each dimension separately. The stride is visualized for a 2D example in Fig. 5.7. The default is a simple stride of 1 in all dimensions. Increasing this number can be a method of down-sampling the input, which is useful when tiny details in the input are not as import for the desired outcome.



(a) Stride of 1 **(b)** Stride of 2

Figure 5.7: Example of different strides used on a 2D input. The green square highlights the first filter position and the orange square the following one depending on the stride. The bigger the stride, the smaller the size of the output filter map.

The padding parameter describes the zero padding of the input data around its borders, as can be seen in Fig. 5.8. It has multiple use cases for CNNs. A particularly notable application is to avoid so-called border effects, but it is also commonly used to simply preserve the spatial size of the input. Additionally, padding is useful in the design of deeper networks as it prevents a fast reduction in the size of feature maps with each layer. Border effects are a phenomenon that arises when the filter is focused on the outer rim of the input. In this case the input on the border only affects the corresponding outer rim of the output. The inner part of the receptive field however, contributes to all the surrounding neighbors of the output too. This means the inner portions of the input effectively get a higher weight, which leads to the network ignoring the border of the input more and more as non-padded layers are stacked. This can be avoided by using padding.

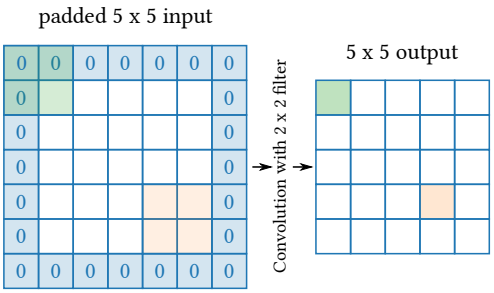


Figure 5.8: Example of zero-padding on the border of a 2D input, showing how the spatial size of the input is preserved in the output feature map. The green and orange shaded areas show where the filters are applied and where the features are then positioned in the feature map.

5.3.2 Cartesian Convolution

Most convolutions in machine learning frameworks are implemented to work with input that follows a Cartesian coordinate system layout. Since the input representation is stored in arrays this means that the input coordinates are restricted to the space of integers, i.e.

depending on the dimension of the convolution

$$\begin{aligned} &\text{1D:} \quad \mathbb{Z} = \{x : x \in \mathbb{Z}\} \\ &\text{2D:} \quad \mathbb{Z}^2 = \{(x, y) : x, y \in \mathbb{Z}\} \\ &\text{3D:} \quad \mathbb{Z}^3 = \{(x, y, z) : x, y, z \in \mathbb{Z}\}\,. \end{aligned} \tag{5.24}$$

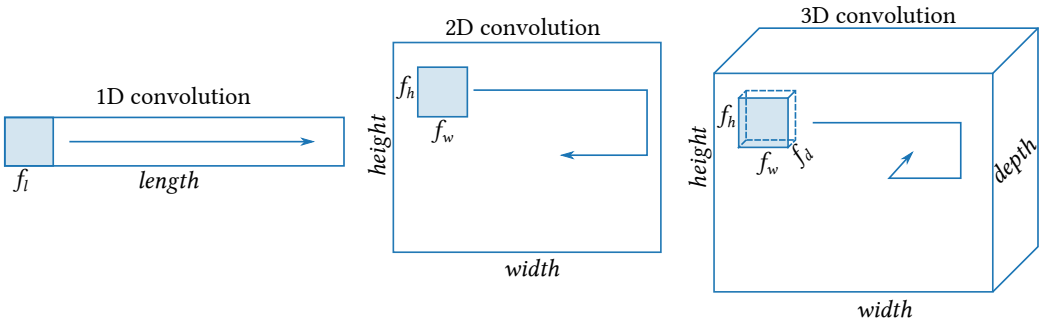


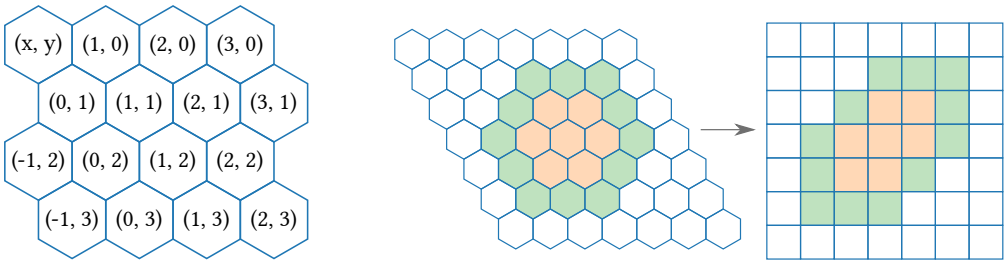
Figure 5.9: Example of input shapes for 1D, 2D and 3D Cartesian convolutions as well as their corresponding filter shapes and how they slide over the input.

Because of this, these types of convolutions are referred to in this thesis as *Cartesian convolutions*. There are various types of Cartesian convolution, with the main difference between them being the above mentioned dimensionality of the input shape, which is typically one, two or three dimensions. The way the filters move across the input depending on dimension of the convolution is illustrated in Fig. 5.9. It is important to note that Cartesian convolutions result in the filters also being rectangular, i.e. Cartesian.

5.3.3 Hexagonal Convolution

A recent approach for DNNs and CNNs has been to utilize symmetries of the input in order to use *group convolutions* [67]. These group convolutions make use of rotations and translations of the filters along the symmetry axes of the input. The advantage of this is that weights can be shared between these filters resulting in a smaller number of hyper-parameters. Mathematically these symmetry axes are defined by so-called *wallpaper groups*. Seventeen of these exist and are used as a classification for repetitive patterns in two dimensions [68], however in the scope of this thesis only the $p6$ group is of interest. For the Cartesian convolutions described above group convolutions would take place within the wallpaper group $p4$ and $p4m$[2]. The $p6$ group provides rotations as multiples of 60°. This allows for the detection of more symmetries in an input, especially if the symmetry of the input is already of a hexagonal shape, as is the case with input data from the SD of the Pierre Auger Observatory. There are several possible options used for coordinate system for hexagonal convolutions [69], but this thesis uses the one shown in Fig. 5.10.

[2]The m is used to denote that mirror reflections are part of the possible symmetries within the group, while the other symmetries in both cases, $p4$ and $p6$, are integer translations



(a) Axial coordinates **(b)** Axial array representation

Figure 5.10: Left: Overview of the axial coordinate systems. Right: Representation of a hexagonal layout in memory. For the axial layout a rectangular filter could be used with the corners masked. Based on [69].

The axial coordinate system is based on the hexagonal lattice structure. In it, any point can be expressed as an integer linear combination $x\vec{e}_1 + y\vec{e}_2$, of two basis vectors $\vec{e}_1$ and $\vec{e}_2$ with an opening angle of 60°. Because convolution filters developed for rectangular images can be adapted to work on the axial coordinate system the easiest, the axial coordinate system is the one most often used for implementing hexagonal convolutions [69], [70]. As can be seen in Fig. 5.10, a hexagonal filter transforms to an almost rectangular, i.e. Cartesian, filter in memory. Their already existing implementations can thus be adapted for use if the corners of the filter are masked during the convolution process. This is the hexagonal convolution technique used in this thesis which was developed and implemented by E. Hoogeboom, et.al. in [69].

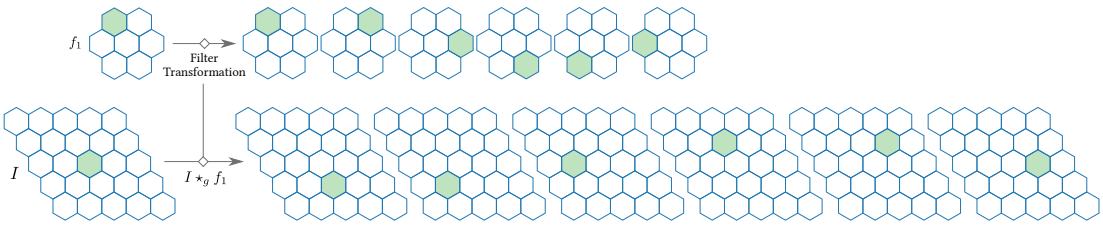


Figure 5.11: Example of the $p6$ group convolution processes for $\mathbb{Z}^2$. The filter f_1 undergoes a filter transformation to all six orientation channels. If applied to an image I the feature map $I' = I \star_g f_1$ likewise consists of six different orientation channels. Adapted from [69].

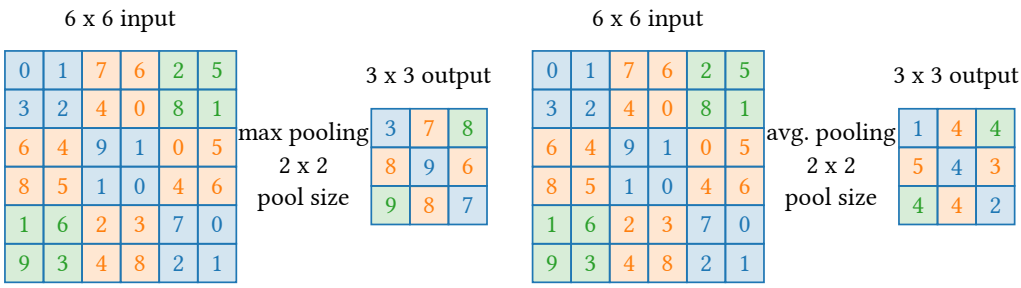
The first step of the hexagonal convolution is always to perform a so-called *filter transformation* for $\mathbb{Z}^2$ with the $p6$ group. This is done by going from one filter f of hexagonal shape to a version of the filter with the six different orientation channels possible within $p6$ symmetry. An example of this can be seen in Fig. 5.11. The resulting feature map, also has 6 orientation channels, from applying said filter to an image I with axial layout via $I \star_g f_1 = I'$. All subsequent convolutions can then take place within the $p6$ group, which then involves both orientation and rotation channel cycling, as described in depth in [69].

5.4 General Layer Types

There are a multitude of layer types that can be used in any NN independent of its category. Here, the three most common layer types will be discussed, as they are also used in this thesis (see chapter 7).

5.4.1 Pooling Layers

As explained in section 5.3, CNNs are used to identify features in the input data, with the feature complexity increasing with network depth. However, one downside to this is, that these convolutions locate the precise position of a feature in the input. This results in the creation of a fully different feature map even for small movements of a feature in the input data. To combat this, a method such as down-sampling the input data can be used. This has the effect of retaining prominent information contained in the input, while blurring out potentially noisy details. The layers used in NNs to perform this task are called *pooling* layers.



(a) Max pooling **(b)** Average pooling

Figure 5.12: Example of a different types of pooling applied to a 2D input. The input data is down-sampled from 6×6 to 3×3 in both cases. The resulting feature map on the right shows the most prominent features for the case of max pooling, while average pooling smooths out the feature map.

The two main types are *average pooling* and *max pooling*. Both types are illustrated in Fig. 5.12. For average pooling, a pool size can be specified and then a window the size of the pool is slid over the input, calculating the average of all values inside the window and saving it in the feature map. The effect of this is the smoothing out of the feature map. Max pooling works similar, the only exception being that inside the window the maximum is calculated instead of the average, which generally has the effect of highlighting the more prominent features in a map. Additionally, *global pooling* layers exist, where either the global average or global maximum of the full input is calculated.

5.4.2 Dense, Flatten, and Reshape Layers

Dense layers are simple fully connected layers and are the most frequently used in deep learning. They can be used to create a fully connected NN, simply by building the network

using only dense layers. However, they are most often used as the last layer in an NN to form the final predictions from the previous, not necessarily fully connected, operations of layers. As such, in the case of a regression task it is commonly applied without any activation function, as the goal is to predict a single real value.

A *flatten* layer is used to, as the name suggests, flatten the input tensor to reduce the dimensionality to one, e.g. performing the transformation

$$(i_\mathrm{h}, i_\mathrm{w}, i_\mathrm{d}) \rightarrow (i_\mathrm{h} \cdot i_\mathrm{w} \cdot i_\mathrm{d})\,. \tag{5.25}$$

It is often used before predicting an output with a dense layer, if a single value prediction is the goal of a regression task.

The *reshape* layer is very versatile, and can be used to shape the input into any form as long as the overall size of the tensor stays the same. For example taking an input tensor size $(i_\mathrm{h} \cdot i_\mathrm{w} \cdot i_\mathrm{d})$, a transformation such as

$$(i_\mathrm{h}, i_\mathrm{w}, i_\mathrm{d}) \rightarrow (i_\mathrm{h}, i_\mathrm{w} \cdot i_\mathrm{d}) \tag{5.26}$$

is possible. With all these building blocks of NNs in mind, the network architecture used in this thesis is developed and described in chapter 7.

6
Simulation Settings and Pre-processing

> *"None of this is real. It is a simulation. We are still on the holodeck."*
>
> — Lt. Commander Data, *Star Trek: The Next Generation*

A crucial step before applying a neural network to any data is to prepare it for read-in by the network. This step is often called *pre-processing* and can refer to the common processes of data augmentation, such as normalizing or re-sizing of the data, or zero padding it. Prior to this the parameters to train the network with have to be chosen. In this thesis, this mainly concerns selecting the parameters of SD events which are expected to contain the information needed to achieve the goal of reconstructing X_{max}.

Additionally, to train a neural network the label, in this case the X_{max} value of an event, has to be known for the network to learn from. Because higher statistics are needed than the FD can provide *Monte Carlo* (MC) simulations are used. While the code to generate AugerPrime detector simulations is now available in $\overline{\mathrm{Off}}\underline{\mathrm{line}}$, it is not yet in a final state regarding the simulation of the trigger. This was in part identified in a study, shown in appendix B, carried out during this research in order to choose which revision of the $\overline{\mathrm{Off}}\underline{\mathrm{line}}$ code should be used to produce the detector simulations needed to train the network. Even though they are still preliminary, the AugerPrime simulations can be used to investigate the expected mass composition sensitivity of AugerPrime.

6.1 CORSIKA Simulation Setup

Before generating AugerPrime detector simulations, the EAS in the atmosphere needs to be simulated separately using the simulation software CORSIKA [71]. Within the Pierre Auger Collaboration, a library of CORSIKA shower simulations is available, commonly referred to as the *Napoli/Praha library*. In this thesis, the simulations use EPOS-LHC [26] as the hadronic interaction model. The available files cover a zenith range of $\theta \in [0°, 65°]$, which is simulated so that the distribution is flat in $\cos^2(\theta)$. The simulations are distributed over an energy range of $E \in [10^{18}\,\mathrm{eV}, 10^{20.2}\,\mathrm{eV}]$, with E^{-1} as the energy spectrum. This is done for 1000 CORSIKA files per $0.1 \log_{10}(E/\mathrm{eV})$ bin, resulting in 22000 CORSIKA files for each of the four simulated primaries: proton, helium, oxygen and iron. In this thesis, unfortunately only the files with an MC energy above $18.5 \log_{10}(E/\mathrm{eV})$ are used. This is because in the subsequent AugerPrime detector simulations with $\overline{\mathrm{Off}}\underline{\mathrm{line}}$, earlier mentioned

short-comings in the trigger simulations especially affect very low energies, where the SD trigger is not fully efficient.

6.2 AugerPrime Simulation and Reconstruction

To create full detector AugerPrime simulations, the $\overline{\text{Off}}\underline{\text{line}}$ Standard Application `SdSimulationReconstructionUpgrade` is used with standard settings. In order to increase statistics, an EAS simulated with CORSIKA can be processed with $\overline{\text{Off}}\underline{\text{line}}$ multiple times. In the first step of creating $\overline{\text{Off}}\underline{\text{line}}$ simulations, a CORSIKA shower core is placed at random point within the SD array. From there, using the CORSIKA generated particle profile at ground the detector response to such a shower can be simulated. The CORSIKA shower can then be "reused" by placing it at a different spot within the array. This is possible because the detector simulations introduce a large amount of randomness by simulating random noise particles or through the differing placement of the shower with respect to the SD stations.

For this thesis, each CORSIKA shower is reused 10 times to create sufficient statistics to train a neural network and this can be done without worrying about overuse of the CORSIKA library. In the next step of the simulation, the *CachedShowerRegenerator* is used to translate the CORSIKA shower to a particle flux on each SD Station. The GEANT4 [72] package is then used to simulate particles and track them through the SSD and WCD. As the particles pass through the detectors, photons from scintillation or water Cherenkov are generated and then further tracked. This is done until the photons either decay or reach a PMT, in which case they are recorded. The number of photoelectrons as a function of time is used as input for the next module which simulates the PMT and electronics responses. The simulated signal traces are then created based on parameterizations from extensive studies of the PMT responses to single photons and noise levels. After these simulation steps, the signals are subjected to the same procedures as real data, i.e. the calibration and trigger criteria described in section 3.2.2. Finally, the event is reconstructed following the procedure mentioned in chapter 4 and both the reconstructed values and the MC truth are stored in ADSTs.

To remove low quality events and failed reconstructions, the simulated ADST files can be further processed with an $\overline{\text{Off}}\underline{\text{line}}$ tool called `selectEvents` which applies quality cuts and merges all files for easier data processing. The `SDcut` file containing the cuts applied for this thesis can be found in appendix C. The most important quality cut applied, is the requirement to keep only events that pass the 6T5 trigger (for a definition see section 3.2.2). This ensures the highest data quality. Additionally, the reconstructed zenith range is restricted to $\theta \in [0°, 60°]$, since it has been shown in chapter 4 that the SSD is not efficient above 60°. Events with reconstructed energies below $10^{18.5}$ eV are also removed, since the MC energy range has a lower limit of $10^{18.5}$ eV. As a result events with

an MC energy lower than $10^{18.5}$ eV are not present and this lowest bin of energies between $10^{18.5}$ eV and $10^{18.6}$ eV is affected by edge effects. Events with a reconstructed energy lower than $10^{18.5}$ eV are lost, while no events from over-reconstructions are gained from a lower bin. This might affect the composition reconstruction of the NN trained in this thesis and as a result, there might be unaccounted for biases in the lowest energy bin. To fix this in the future, MC simulations from the lower bin of $10^{18.4}$ eV should also be used and the cut on reconstructed energies should be applied afterwards. However, this error was identified too late in this thesis to correct for it in the results presented here. No upper limit is set on the reconstructed energies, as there the reconstruction is generally very robust due to high station counts.

6.3 Pre-processing Detector Simulations with Python

The merged ADST files are read-in using the Python tool PyIK mentioned in section 3.5.2, to extract the parameters needed to train the neural network. The subsequent pre-processing and data augmentation algorithms are performed with Python as well.

6.3.1 Parameter Selections and Read-out

To reconstruct X_{max}, primarily the SSD and WCD detector responses will be used. Additionally, a few high-level parameters, which can be obtained with high precision from the SD reconstruction, are used to focus the network on X_{max}. Following this plan, the following parameters are selected as input for the DNN:

- the calibrated signal traces $S(t)_{\mathrm{SSD}}$ and $S(t)_{\mathrm{WCD}}$ (mean over all three WCD PMT traces)
- the arrival time at the individual stations t_i
- the SD reconstructed energy $E_{\mathrm{SD,rec}}$
- the SD reconstructed zenith angle $\theta_{\mathrm{SD,rec}}$
- the MC depth of shower maximum $X_{\mathrm{max,MC}}$ as the label

The two detector traces are the most integral part in this study. The goal is to reconstruct the shower maximum on an event-by-event basis primarily based on the differences between the signal traces recorded by the two detector types and to study the improvement in composition sensitivity gained with the addition of the SSD. The arrival time, zenith, and energy are added, as they are the most descriptive high-level parameters in an event and thus are likely useful to determine the X_{max} of an event. For training and predicting, only SD reconstructed energies and zeniths are used, since the MC information is not available

in real data. The distributions of the event-level parameters $E_{\mathrm{SD,rec}}$, $X_{\mathrm{max,MC}}$, and $\theta_{\mathrm{SD,rec}}$ can be seen in Fig. 6.1. $E_{\mathrm{SD,rec}}$ and $\theta_{\mathrm{SD,rec}}$ are read out and used as is.

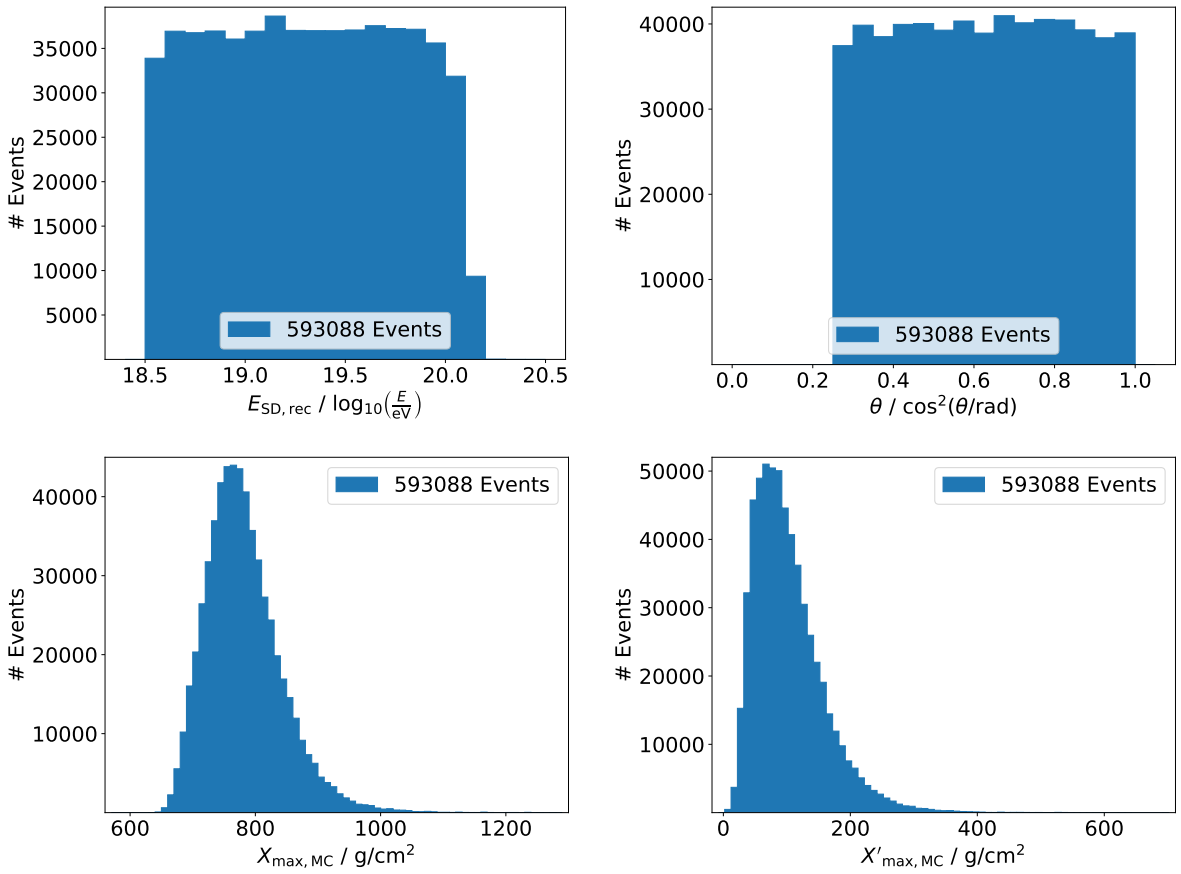


Figure 6.1: Full data set distributions. Top left: The $E_{\mathrm{SD,rec}}$ distribution is flat in E^{-1}. The earlier mentioned effects of bin migration and edge effects are visible in the highest and lowest bins. Top right: Zenith angle distribution where $\theta_{\mathrm{SD,rec}}$ is plotted as a function of $\cos^2(\theta_{\mathrm{SD,rec}})$ to show the flatness. Bottom: MC X_{max} distribution to be used as the truth value $X_{\mathrm{max,MC}}$ before (left) and after (right) accounting for the elongation rate.

6.3.2 Data Augmentation and Elongation Rate Correction

In order to achieve a good network performance, the SSD and WCD traces as well as the arrival times and $X_{\mathrm{max,MC}}$, are augmented before being used for training. For example when studying the pure-proton and pure-iron lines of the three hadronic interaction models shown in Fig. 2.6, it is evident that X_{max} values are correlated with energy. This effect is known as the *elongation rate*. This connection between X_{max} and energy is something the neural network would learn. This could lead to the trained model predicting X_{max} values solely based on the reconstructed energy of an event. Such a behavior is not desired, as it would direct the network away from forming predictions based on differences in the SSD and WCD traces. This energy dependence can be removed by subtracting the pure-iron

elongation rate as implemented in EPOS-LHC [26] from every $X_{\mathrm{max,MC}}$ value as follows

$$X'_{\mathrm{max,MC}} = X_{\mathrm{max,MC}} - \left(\underbrace{600\,\frac{\mathrm{g}}{\mathrm{cm}^2}}_{\substack{\text{additional}\\ \text{scaling}\\ \text{factor}}} + \underbrace{63.1\,\frac{\mathrm{g}}{\mathrm{cm}^2} \cdot (E_{\mathrm{SD,rec}} - 18) + 1.97\,\frac{\mathrm{g}}{\mathrm{cm}^2} \cdot (E_{\mathrm{SD,rec}} - 18)^2}_{\substack{\text{EPOS-LHC elongation rate line}\\ \text{for iron}}} \right) . \tag{6.1}$$

In addition to the EPOS-LHC line, $600\,\mathrm{g/cm^2}$ are subtracted in order to scale $X_{\mathrm{max,MC}}$ down to smaller values, while keeping all values seen in simulation positive. The 18 in eq. (6.1) stems from the $\log_{10}$ of the EPOS-LHC reference energy of 1 EeV (10^{18} eV). This provides an additional benefit, as this adjustment shifts the $X_{\mathrm{max,MC}}$ values to a smaller scale. Hereafter, $X_{\mathrm{max,MC}}$ is simply used to refer to the adapted values $X'_{\mathrm{max,MC}}$.

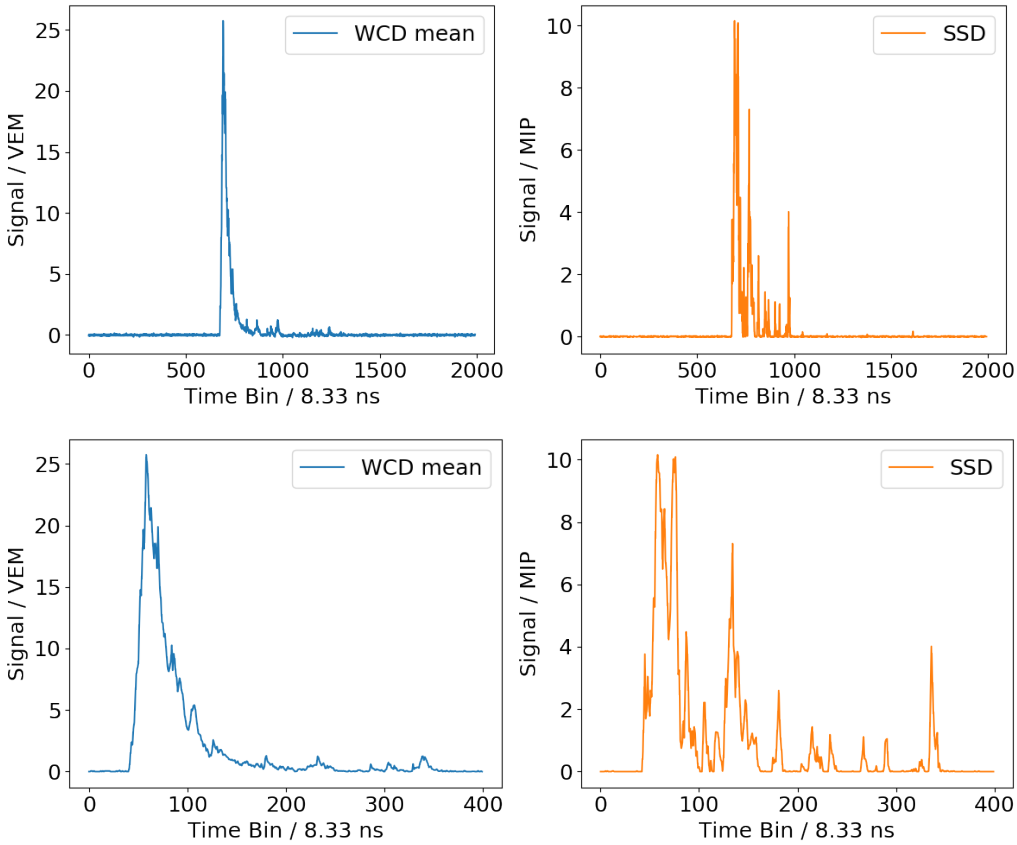


Figure 6.2: Top row: WCD (left) and SSD (right) traces before pre-processing, showing the full trace of 2048 time bins. Bottom row: WCD (left) and SSD (right) traces after removing the leading zeros up 48 bins before the trigger and the empty bins at the and of the trace, keeping only the relevant signal.

The signal traces $S(t)_{\mathrm{SSD}}$ and $S(t)_{\mathrm{WCD}}$ have a full length of 2048 bins, with a bin width of 8.33 ns each. The length of the actual signal contained in these traces is generally

much shorter and therefore the signal traces contain a lot of empty bins, as can be seen in Fig. 6.2. This is especially true for a fixed number of bins at the start of each trace, which originates from the way the traces are read out when a trigger is received. In the Offline trigger simulation, this is characterized by the so-called *latch bin*, which indicates the number of bins before the actual signal starts. For UB stations it has a value of 246. For upgraded stations with a UUB this is internally multiplied by a factor of three, to account for the additional bins. The resulting 738 empty bins are unnecessary information for this analysis and are therefore removed, with a small buffer of $16 \cdot 3$[1] bins. This results in 690 bins being cut from the start of the trace. The next 400 bins of trace data are kept for both detectors and the remaining 958 bins at the end of the trace are also cut off. On average the full trace information should be contained, as 400 bins represents a generous thickness of shower front of roughly $400 \cdot 8.33\,\text{ns} \cdot c \approx 1\,\text{km}$ passing through the detector in that time. Additionally, for the WCD case the mean over all three PMT traces is calculated, to have on input to compare the SSD trace to.

The arrival time t_i, in nanoseconds, for each station participating in an event is also transformed to the relative arrival time

$$\Delta t_i = \frac{t_i - t_{\text{min}}}{8.33\,\text{ns}}\,, \tag{6.2}$$

where t_{min}, also in nanoseconds, is the earliest of all the arrival times in an event. The division by the trace bin width of 8.33 ns is performed to adjust the arrival times from an absolute value of large magnitude $\mathcal{O}\,(10^9\,\text{s})$ to a small relative arrival time in units of trace bins.

6.3.3 Axial Layout and Padding

The network developed in this thesis is supposed to learn from a combination of the SSD and WCD trace information, as well as the stations relative position to each other. Since the number of stations varies with each event, and the input shape for the neural network has to be the same for all events, it is important to unify the station layout in pre-processing. In Fig. 6.3 an example of a shower footprint on the ground before pre-processing can be seen.

For this thesis, the neural network was designed to process the event data in a 9×9 grid. The first step of bringing the input data into this shape, is to create an array, A_{ideal}, of station positions, $\vec{x}_{\text{ideal},j}$, which are centered around the point (0 km, 0 km) with a station spacing of 1500 m and an axial layout. Then, arrays of shape $9 \times 9 \times d$ are created and initialized with zeros, where d is the dimension of the respective future inputs, a list of which can be seen in Tab. 6.1.

[1] This was set as an offset of 16 bins from the UB latch bin value and is then adapted to the UUB

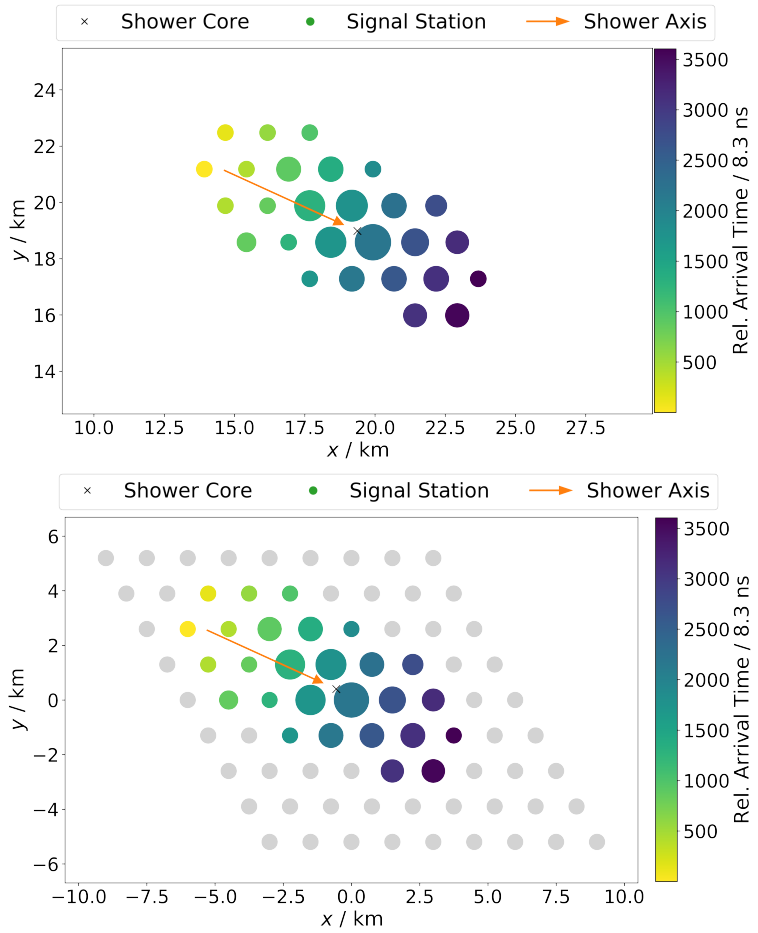


Figure 6.3: Top: Original shower footprint of candidate stations. The size of each station represents the size of the total signal in the SSD, while the color represents the relative arrival time, calculated as mentioned above. Bottom: The same footprint after pre-processing with the gray dots representing the zero padding stations to fill in the axial mapping needed for the neural network. All stations are now centered around the hottest station, which is located at (0 km,0 km).

Table 6.1: Input arrays for the DNN to train on and their respective dimensions and units. X_{max} is separated as it serves as the label that needs to be learned by the network.

Input	Dimensions	Unit
SSD traces	array(9, 9, 400)	MIP
WCD traces	array(9, 9, 400)	VEM
Energy, zenith, time	array(9, 9, 3)	$\log_{10}\left(\frac{E}{\mathrm{eV}}\right)$,rad,trace-bins
$X_{\mathrm{max,MC}}$	array(1)	g/cm^2

The input arrays are filled with the necessary data by looping over the station vector, which contains all stations that had a any signal at the time of the event. Station information is only read out under the condition, that the current station in the loop is marked as a *candidate station*. A candidate station is a station that was selected by the

trigger, simulated or real, as part of the event. Only candidate stations are used as this can largely filter out random noise stations, which are also included in the station vector. If the current station is flagged as a candidate, its trace information and the arrival time are read out and adapted as described above. The station position $\vec{x}_i$ is also saved.

In each event the station with the highest signal in the WCD is flagged as the *hottest station*. Its coordinates $\vec{x}_{\mathrm{H}}$ are then used as the reference point for all other stations contained in the event and their relative position to the hottest station is calculated as

$$\vec{x}_i' = \vec{x}_i - \vec{x}_{\mathrm{H}} \,. \tag{6.3}$$

The hottest station is also included which places it at the center of the array (0 km, 0 km). These new coordinates are then used to calculate the placement of the stations in A_{ideal}. This is done by calculating the euclidean distance d_{euclid} of $\vec{x}_{\mathrm{i}}'$ to each ideal station position $\vec{x}_{\mathrm{ideal},j}$ in A_{ideal}. The ideal station with the minimal distance to $\vec{x}_i'$ represents the ideal position of that station i:

$$\min\left(d_{\mathrm{euclid}}(\vec{x}_i', \vec{x}_{\mathrm{ideal},j})\right) = \min\left(\sqrt{\sum_k \vec{x}_{i,k}' - x_{\mathrm{ideal},j,k}}\right) \tag{6.4}$$

The index of this ideal station in A_{ideal} is then used to write the traces and arrival times to the correct place in their respective input arrays, replacing the zeros. The relative arrival times t_i are combined with $E_{\mathrm{SD,rec}}$ and $\theta_{\mathrm{SD,rec}}$ into one array. Saving the energy $E_{\mathrm{SD,rec}}$ and zenith angle $\theta_{\mathrm{SD,rec}}$ values for each station seems excessive, considering they are event-level variables. However, this is done to later ease their combination with the station-specific data.

The results of this procedure are zero padded arrays that contain the necessary input information at the relative station positions. The footprint of the padded event data can be seen in Fig. 6.3 at the bottom. The Python package NumPy [73] is used for the majority of these operations and the input arrays are saved as compressed NumPy array files `.npz` for later used with the DNN described in chapter 7.

6.3.4 Removal of Geometric Degeneracy

In addition to the data augmentations described earlier, a method was developed for this thesis with the goal of improving the performance and speed of the training of the neural network. Since the number of stations participating in an event varies, a rather large grid would be needed to contain the entire footprint. On top of that, large amounts of data are filled with zeros, from the zero-padding, which need to be processed by the network, especially for events with lower energies or zeniths and therefore smaller footprints. Processing the input data as is, means that the positions of these “inactive stations” changes

a lot in almost every event. As a result of this the network needs a lot of nodes and a long time to converge.

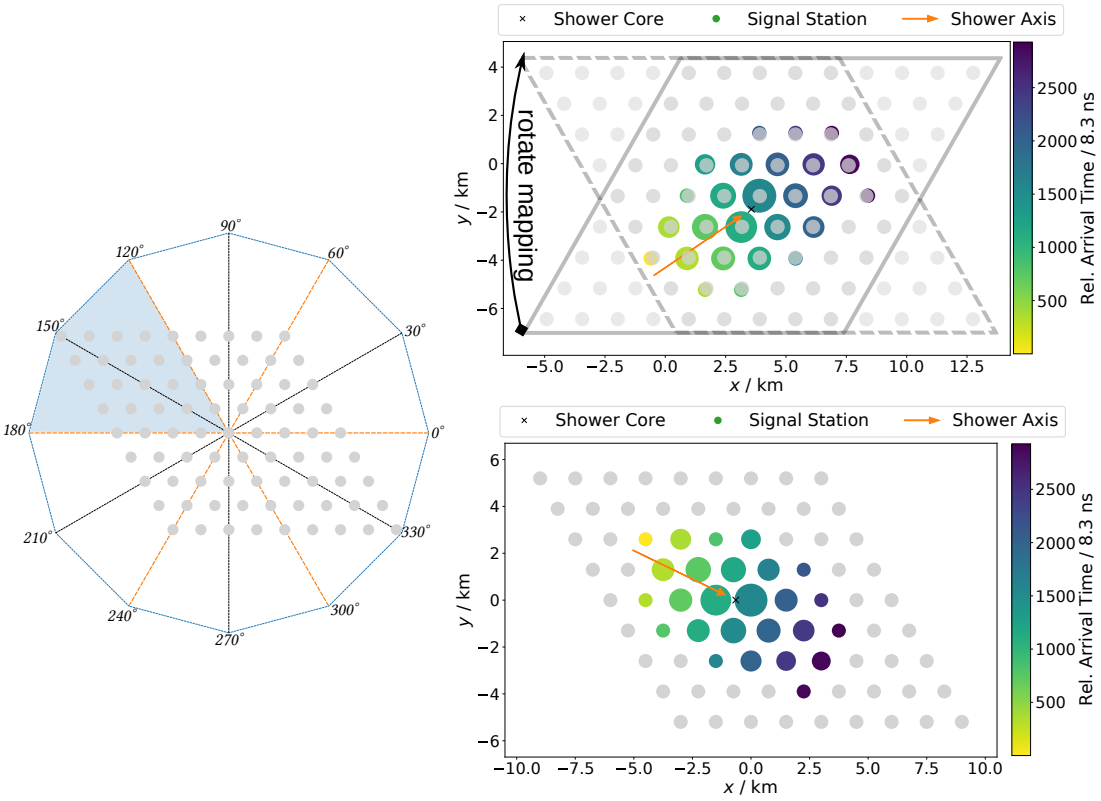


Figure 6.4: Left: All events are rotated to align with the $\phi_{\mathrm{SD,rec}} = 150°$ axis, indicated by the blue wedge. Top right: An example event is shown with the initial orientation of the station mapping grid indicated by a solid gray line. The dashed gray line indicates a rotated version of the mapping that contains all stations in the event. Bottom right: The final event that is used for training.

To combat this, a rotation of the mapping was developed, which results in the positions of "active station" varying less from event to event. The network then "sees" the inactive stations more commonly at the same position in the input array and can adapt the activation of nodes accordingly. This technique also allows for the total number of nodes in the network to be reduced. This is carried out by rotating the mapping of for each individual event so that the azimuth angle of each event always aligns with the same axis of the mapping grid. As can be seen in Fig. 6.4, the north-west/south-east azimuth axis of the 9×9 grid covers only a distance of $\approx 12\,\mathrm{km}$, while the south-west/north-east axis covers a much larger distance of $\approx 21\,\mathrm{km}$. Thus, the long axis of the axial grid is chosen as the axis to which each event is rotated to align with. This is indicated in Fig. 6.4 on the left by the blue wedge. With this last step the AugerPrime simulations are ready to be used as an input for the DNN.

7

Reconstruction of X_{max} from AugerPrime Simulations

> *I am hopeful that one day I will discover my own humanity. Until then, I will continue learning, changing, growing, and trying to become more than what I am.*
>
> — Lt. Commander Data, *Star Trek: The Next Generation*

Once the simulation data has passed the pre-processing stage, it has to be transferred to the local GPU computing cluster for data-splitting and training. Of the full data set of 593088 events, $\sim 70\,\%$ are taken as the training set. From the remaining 177024 events another split of roughly $\sim 70\,\%/30\,\%$ is performed. The split off $30\,\%$ are used as the validation set and the $70\,\%$ are used as the test set. The resulting event statistics after splitting are summarized in table table 7.1.

Table 7.1: The number of events in the training, validation, and test set are summarized on the left. The general settings used to train the networks are shown on the right.

Data set	# Events	NN parameters	Settings
Training	416064	Batch size	32
Validation	58976	Learning rate	0.001
Test (proton)	30240	Epochs	150
Test (helium)	29312	Optimizer	Adam
Test (oxygen)	28512	Weight/bias initializer	HeNormal
Test (iron)	29984		

Two networks are trained with exactly the same data sets and splitting and utilize the same network architecture. One network uses both SSD and WCD traces, while the other is trained on WCD-only data, hereafter be referred to as $M_{\mathrm{SSD+WCD}}$ and M_{WCD}, respectively. This is done so as to directly compare the two networks in order to evaluate the improvement achieved for event-by-event composition sensitivity by the addition of the SSD on top of the WCD. Both models share the same general settings that are listed in Tab. 7.1. Since the reconstruction of a continuous value, here X_{max}, is a regression task, HeNormal was chosen as the initializer for weights and biases. Adam was chosen as the optimizer and is used with the recommended default parameters mentioned in section 5.2.4. Because the goal of this thesis is studying the composition sensitivity of

AugerPrime, the metric

$$\text{metric} = \sqrt{\text{Var}(\vec{y}_{\text{true}} - \vec{y}_{\text{pred}})} \tag{7.1}$$

is chosen. This metric calculates the variance and its square root over each epoch and serves as a close indicator of the X_{max} reconstruction resolution. Previous iterations of the networks presented here showed signs that longer training to achieve better results, so a total training duration of 150 epochs was chosen. For each epoch the data is processed in batches of 32, as that is around the maximum amount that the local GPU computing cluster can fit into memory at once.

7.1 Network Architecture

The network architecture developed for this thesis matches the shape of the input data described in the chapter 6 and is visualized in Fig. 7.1.

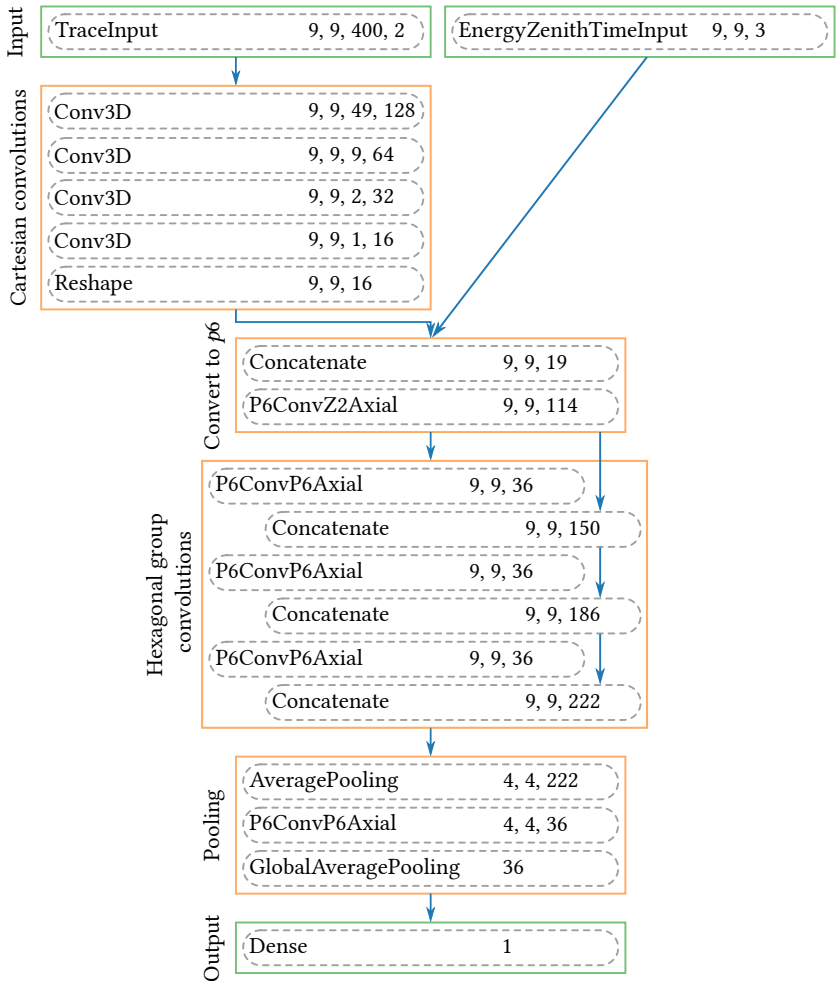


Figure 7.1: Schematic view of the network architecture used to reconstruct X_{max} with AugerPrime simulations. Each rounded box represents a single layer, with the output shape noted in the right part of the box. The rectangular boxes indicate the different groups of operations.

All code for this is written using Keras [74] with TensorFlow [75] as a back-end. Apart from the input and the output layer, it can be divided into three different categories, the Cartesian trace convolutions, the hexagonal convolutions (including the transformation to $p6$ as well as subsequent group convolutions), and the final layers.

7.1.1 3D Cartesian Trace Convolution

As explained in section 3.3, the main composition information of AugerPrime should be found in the differing responses of the two detector types to the EAS components. Thus, in the first part of the network architecture the two traces $S(t)_{\mathrm{SSD}}$ and $S(t)_{\mathrm{WCD}}$ are used as input for a sequence of Cartesian convolutions. Even though a 3D convolution layer structure is used, the actual convolution is only carried out along the trace axis of each SD station individually. This is done by setting the filters shape for each layer to have the shape $(1,\, 1,\, f_{\mathrm{depth}})$. The 3D convolutions are used to simply and efficiently retain the spatial information, meaning the station's relative positions to each other within the 9×9 grid. The outcome of each convolution layer is a $(9,\, 9,\, out_{\mathrm{depth}},\, n_{\mathrm{features}})$ feature map, where out_{depth} depends on how often the filter size fits into the input size. The last dimension represents the set of features extracted from each individual SD station that contain most of the composition information.

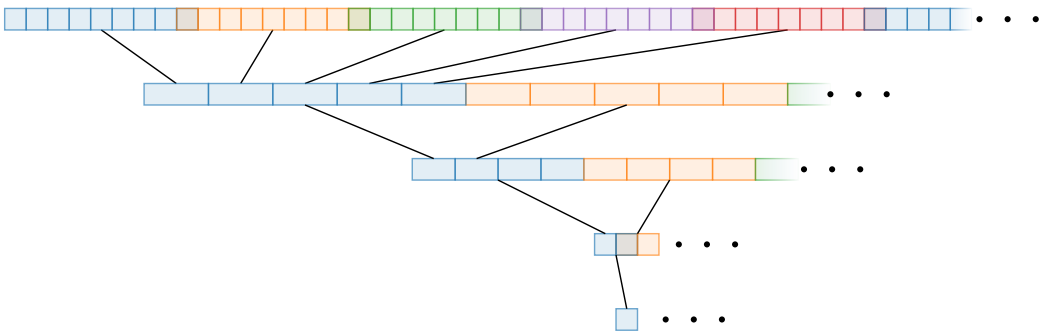


Figure 7.2: Schematic view of the convolution layers focusing on one station (the two channel structure is left out for ease of visibility). The individual filter windows in a layer are represented by the boxes having the same color, while the number of boxes indicate the size of the filter window. In the first layer the filter windows overlap in order to scale the size of the traces from 400 bins.

The comparison between the two trace types is implemented in this thesis by treating each of the trace pairs as individual channels of the same input. This is indicated by the 2 in the last dimension of the trace input in Fig. 7.1. The first step of the trace convolutions is to take a relatively wide filter window of 9 time bins as well as a stride of 8 and apply it along the time axis and over the SSD and WCD channels. The goal of this is to identify temporally short features in the traces and compare the differing responses of the two detectors with a large enough window width for a particle to travel between the

detectors. The next layers take the progressively shorter feature maps from each of these two detector "snapshots" and compare them to the feature maps from their neighboring snapshots. This is continued until a single set of 16 features containing the most important composition information of each station is identified. This process is illustrated in Fig. 7.2. The final result of these Cartesian convolutions, after the application of a reshape layer, is a feature map of shape (9, 9, 16).

7.1.2 2D Hexagonal Group Convolution

In a next step this feature map is concatenated with the additional input information of $E_{\mathrm{SD,rec}}$, $\theta_{\mathrm{SD,rec}}$, and Δt_i, resulting in a combined feature map of shape (9, 9, 19). Hexagonal group convolutions are then utilized to perform convolutions on the actual station grid. This is done to extract features from the full event by comparing the different responses to the EAS between the stations. As explained in section 5.3.3, to process the (9, 9, 19) feature map further with hexagonal convolutions, the initial $\mathbb{Z}^2$ to $p6$ conversion layer has to be applied. This layer is applied with a filter of size 1, which is equivalent to one single station, to convert the feature maps of every station individually. The result of this is an output of shape of $(9, 9, 19 \times 6 = 114)$.

Afterwards, the convolutions within the $p6$ group are performed on this input tensor. The approach taken in this thesis is inspired by the densely connected convolutions described in [76]. A NN can be deeper and more efficient if in each convolution layer it has both a short connection between layers and a short connection to the input and the output. A structure like that results in a reuse of the important original information which is intended to stabilize the network [76]. In the analysis presented, these short connections between input and output are achieved by concatenating the output of each $p6$ group convolution layer again with its input, preserving the input and the information in the last convolution for the next $p6$ group convolution. All of these convolution steps use a padding to sustain the (9, 9) shape of the first two dimensions throughout the hexagonal convolution block. This is needed to make the concatenations with previous layers output possible. The filter size in these convolutions was chosen as 3, which represents a filter that covers exactly one SD hexagon, i.e. 7 stations.

7.1.3 Final Layers

The last layers, are a combination of a last hexagonal convolution, as well as two pooling layers. Average pooling is applied in order to smooth out the feature map and reduce the size. For the second pooling layer global average pooling is chosen, in order to combine all the features into a 1D feature map. Finally, a dense layer is used to combine all the features into one single output, resulting in the prediction of X_{max}.

The total number of free parameters the network has to train differs slightly between the

two versions, $M_{\mathrm{SSD+WCD}}$ and M_{WCD}. The reason for this is only the number of channels used when performing the trace-wise convolutions. As mentioned earlier, the two detector traces are treated as two different channels of the same input, but this can only be done if both detector traces are used. The two options available to train the WCD-only network are to either set all SSD traces to zero, or to remove the SSD trace altogether and use only one channel. The second option is chosen for this thesis in order to increase efficiency and speed up the training of the neural network, as there are be slightly fewer parameters to train this way (89449 for $M_{\mathrm{SSD+WCD}}$ vs. 88297 M_{WCD}).

7.2 In-training Data Augmentation

Most augmentation of the data is performed during the pre-processing stage and is described in detail in section 6.3.2. There are, however, two methods of data augmentation utilized in the work for this thesis, that are applied during the training process. These are event weighting and the simulation of non-functioning stations.

7.2.1 Re-weighting the $X_{\mathrm{max,MC}}$ Distribution

In Fig. 7.3 the combined $X_{\mathrm{max,MC}}$ distribution of the training and validation set is shown. As can be seen this distribution is not flat in X_{max}. Because of this, the tails of the distribution, with both deeper and shallower X_{max} values, are under-represented during training. As a result the network learns how to predict the mean of the $X_{\mathrm{max,MC}}$ distribution best. In order to prevent this behavior, weights can be applied to every event during training. Higher weights are assigned to the tails of the distribution where the $X_{\mathrm{max,MC}}$ statistics are much lower, flattening the distribution. This way, the network regards those events as having a higher importance, pulling predictions away from the mean of the distribution.

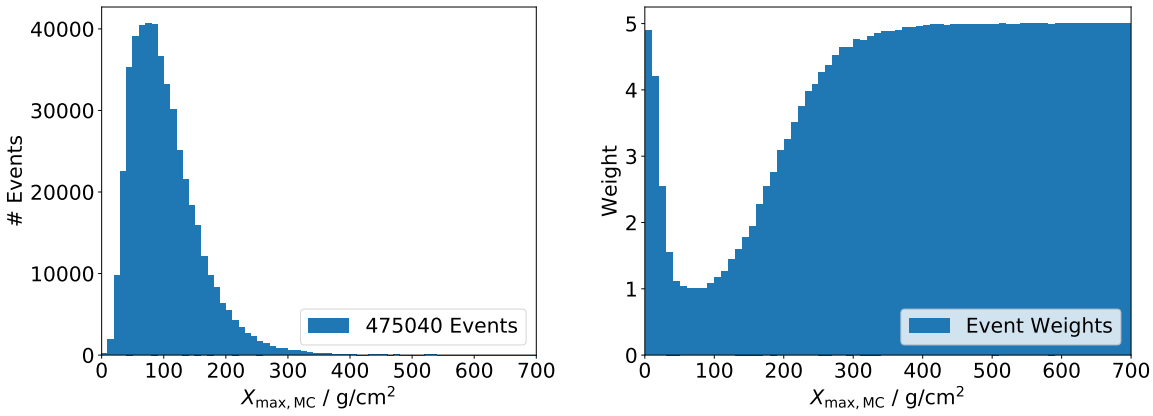


Figure 7.3: Left: Input $X_{\mathrm{max,MC}}$ distribution. Right: Calculated corresponding weights.

In this thesis, a simple way to calculate weights is chosen. The weights are based on

the event counts in the $X_{\mathrm{max,MC}}$ histogram H and calculated for each bin in H via

$$w_i = \frac{1}{\left(\frac{H_{\mathrm{counts,i}}}{H_{\mathrm{counts,\ max}}} \cdot \left(1 - \frac{1}{w_{\mathrm{max}}}\right) + \frac{1}{w_{\mathrm{max}}}\right)}\,, \tag{7.2}$$

where $H_{\mathrm{counts,i}}$ is the number of events in each bin i and $H_{\mathrm{counts,\ max}}$ is the maximum bin content of H. This results in the weight histogram, shown in Fig. 7.3, which importantly has the same bin width as H. The possible weights cover a range of $w \in [1, w_{\mathrm{max}}]$, so that $X_{\mathrm{max,MC}}$ values from a bin with high statistics in H get assigned a weight $\simeq 1$ and events with an $X_{\mathrm{max,MC}}$ value from the flanks of the distribution gets a weight of $1 < w \leq w_{\mathrm{max}}$. For this thesis, the maximum weight of $w_{\mathrm{max}} = 5$ was chosen, in order to avoid a too extreme value, and performed well on an earlier version of the networks. This value, however, could certainly be tweaked in future iterations to fine-tune the performance. During training, for each event a weight is drawn from this distribution corresponding to the bin of its $X_{\mathrm{max,MC}}$ value. This is fed into the network together with the input information and applied as a weight when calculating the cost function.

7.2.2 Non-functioning Station Simulation

$\overline{\mathrm{Off}}\underline{\mathrm{line}}$ simulations do not necessarily represent the real data taken at the Observatory. In the detector simulation an ideal detector is assumed, but in reality not all stations in the SD array are always functioning. This can be due to various reasons, for example stations are considered non-functioning when at least two of the three PMTs in the WCD are not working properly. The number of non-functioning SD stations is a property that is monitored constantly and made available [77]. At the time this research was carried out, the average rate of non-function stations in the SD array was seen to be 25. From there, this value is used to calculate the probability for any station to be non-functioning of

$$p = \frac{25}{1600} = \frac{1}{64} = 0.015625\,. \tag{7.3}$$

This probability is then used to randomly simulate non-function stations for each event during training. The idea behind this is two-fold. On the one hand it serves as a means to have a better representation of real AugerPrime data, so that once enough statistics with the upgraded SD are available, the model trained for this thesis $M_{\mathrm{SSD+WCD}}$ could be applied to data. On the other hand it is a very useful method to bootstrap the data the neural network "sees" during training by essentially adding random noise to the simulation data, which can help stretch training statistics significantly.

At first, a random number $R \in [0, 1]$ is drawn from a uniform distribution. This random number is compared to the binomial probabilities of having $k = 1, 2, 3,$ or 4 non-functioning stations in the available grid of $n = 9 \times 9 = 81$ stations. This is obtained by first taking

into account the aforementioned probability p, where

$$\Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k} \tag{7.4}$$

is the probability that an exact number of $X = k$ stations is non-functioning. The amount of non-functioning stations $\mathrm{SD}_{\text{non-func.}}$ in an event is then chosen as

$$SD_{\text{non-func.}} = \begin{cases} 0, & R > \sum_{k=1}^{4} \Pr(X=k) \\ 1, & \sum_{k=2}^{4} \Pr(X=k) < R \leq \sum_{k=1}^{4} \Pr(X=k) \\ 2, & \sum_{k=3}^{4} \Pr(X=k) < R \leq \sum_{k=2}^{4} \Pr(X=k) \\ 3, & \Pr(X=4) < R \leq \sum_{k=3}^{4} \Pr(X=k) \\ 4, & R \leq \Pr(X=4)\,. \end{cases} \tag{7.5}$$

The maximum of four non-functioning stations is chosen, because the probability of having five or more such stations in a grid of size 81 is below 1 %. Once the amount of $SD_{\text{non-func.}}$ is calculated, the actual stations for that event are that are disabled are chosen at random from the 81 possible stations and their trace data is set to zero. This process is carried out for every event each time it is processed during training, i.e. every epoch, and therefore a different random set of non-functioning stations is selected each time the event is seen.

7.3 X_{max} Reconstruction Results with AugerPrime Simulations

Once both networks are fully trained, they are evaluated according to the same criteria in order to compare the performance with and without SSD trace information. The models are saved after each epoch and out of the 150 available files for both $M_{\text{SSD+WCD}}$ and M_{WCD}, the best model file to use for evaluation has to be chosen.

7.3.1 Evaluation Criteria

Both trained models are evaluated based on the following two aspects

1. The correlation between $X_{\text{max,MC}}$ and the predicted value, hereafter referred to as $X_{\text{max,DNN}}$

2. The overall resolution $\sigma\left(\Delta X_{\text{max,DNN-MC}}\right)$ and the composition dependent reconstruction bias $\langle \Delta X_{\text{max,DNN-MC}} \rangle$.

To evaluate the correlation between $X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$ the correlation coefficient C_{corr} is calculated via

$$C_{\mathrm{corr}} = \frac{\sum_{i=1}^{n} \left(X_{\mathrm{max,MC,i}} - \bar{X}_{\mathrm{max,MC}}\right)\left(X_{\mathrm{max,DNN,i}} - \bar{X}_{\mathrm{max,DNN}}\right)}{\sqrt{\sum_{i=1}^{n} \left(X_{\mathrm{max,MC,i}} - \bar{X}_{\mathrm{max,MC}}\right)^2}\sqrt{\sum_{i=1}^{n} \left(X_{\mathrm{max,DNN,i}} - \bar{X}_{\mathrm{max,DNN}}\right)^2}}\,, \tag{7.6}$$

where $X_{\mathrm{max,MC,i}}$ and $X_{\mathrm{max,DNN,i}}$ are individual data points in the two datasets ($X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$) to be compared, and $\bar{X}_{\mathrm{max,MC}}$ and $\bar{X}_{\mathrm{max,DNN}}$ are the respective means of these. A line fit to the correlation is also obtained to compare to the bisecting line.

To calculate the reconstruction bias and resolution, all $X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$ are split by their corresponding event energy and primary type. This is done in bins of size $0.1\log_{10}(E/\mathrm{eV})$ in the range of [18.5, 20.0]. All events where $E_{\mathrm{SD,rec}} > 20.0\log_{10}(E/\mathrm{eV})$ are grouped into one bin with bin-center $\langle E_{\mathrm{SD,rec}}\rangle$ to avoid having too low statistics at the highest energies, a practice which is commonly applied to data. For each of these bins

$$\Delta X^i_{\mathrm{max}} = X^i_{\mathrm{max,DNN}} - X^i_{\mathrm{max,MC}} \tag{7.7}$$

is calculated, where i is the primary type (p, He, O, Fe). The distribution $\Delta X^i_{\mathrm{max}}$ for each energy bin and composition is then simply fit with a Gaussian of the form

$$\frac{A}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}\,, \tag{7.8}$$

where A is a scaling factor for the amplitude of the distribution. The important parameters for this analysis are μ and σ as they directly represent the reconstruction bias $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}}\rangle$ and resolution $\sigma(\Delta X_{\mathrm{max,DNN\text{-}MC}})$, respectively.

7.3.2 Choosing the Epoch of Evaluation

The cost and metric curves show the training and validation results of the respective functions, calculated at the end of each epoch. These curves are displayed for the full training period of 150 epochs for both networks in Fig. 7.4.

From these, the epoch at which the network is evaluated has to be chosen. Both the training and the validation cost of $M_{\mathrm{SSD+WCD}}$ are consistently lower than that of M_{WCD}. While the training cost decreases smoothly over time in both cases, the validation cost shows strong fluctuations. One possible reason for this could be that the validation set is not large enough to represent the data well. However, the distributions of the parameters ($E_{\mathrm{SD,rec}}$, $\theta_{\mathrm{SD,rec}}$, and $X_{\mathrm{max,MC}}$) shown in Fig. 6.1 were also checked for each subset (train, validation, test) individually to verify, that they show the same behavior within expected statistical fluctuations.

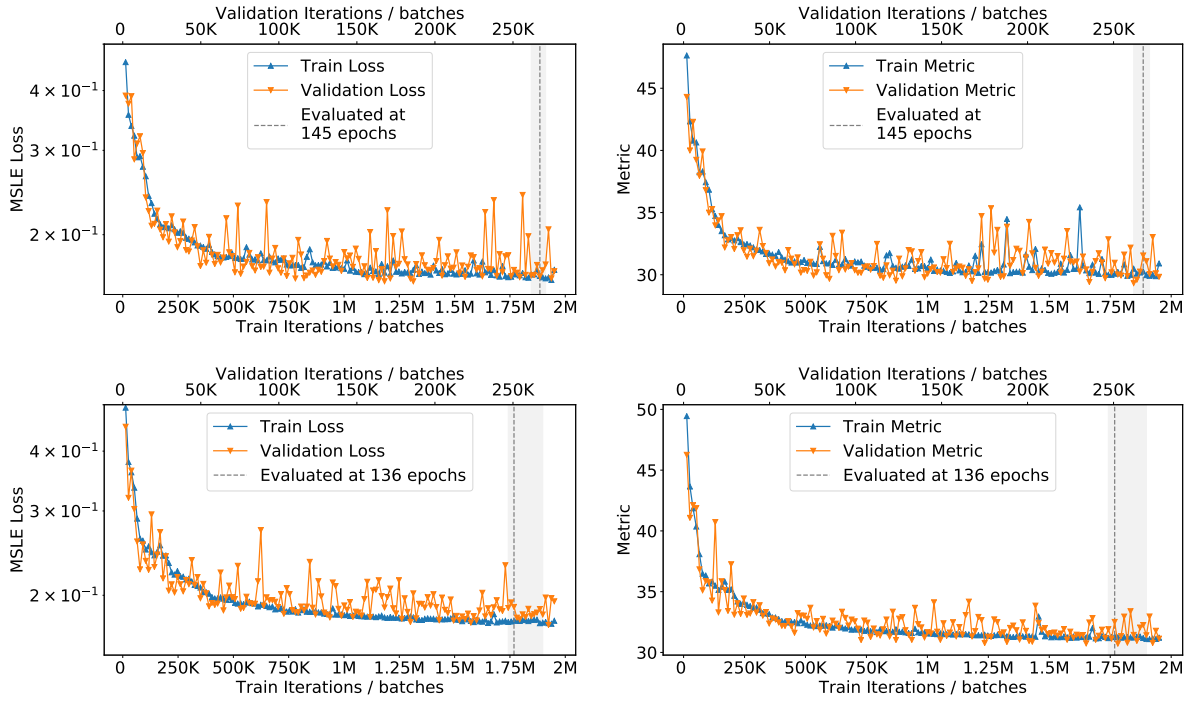


Figure 7.4: Left: Training (blue, up-triangle) and validation (orange, down-triangle) cost for the full training time of 150 epochs for $M_{\mathrm{SSD+WCD}}$ (top) and M_{WCD} (bottom). The data points indicate the evaluation at the end of the epoch, while the number of iterations is given in units of batches. The vertical line (gray, dashed) indicates the evaluation epoch (145 for $M_{\mathrm{SSD+WCD}}$ and 136 for M_{WCD}) and the shaded regions signifies the window from which it is chosen. Right: Training and validation metric.

The fluctuations are, in general, stronger in $M_{\mathrm{SSD+WCD}}$ than in M_{WCD}. The SSD traces could in theory be the cause and these fluctuations might lessen once the Offline simulations are finalized. This behavior makes it somewhat more challenging to choose the best epoch for evaluation. Since the training cost consistently decreases, and there is no sign of over-training visible in either case, the later epochs are those considered for evaluation. There are small regions, marked by the shaded bands in Fig. 7.4, with lower fluctuations in the validation cost. These bands are defined as the evaluation window, and the predictions are run for all of those epochs. The prediction results are evaluated as described earlier, with a focus on the resolution and minimizing the composition dependent bias. The models with the best performance on the test data set in this evaluation window, corresponding to epoch 145 for $M_{\mathrm{SSD+WCD}}$ and 136 for M_{WCD}, are chosen for further analysis and presentation.

7.3.3 Evaluating the Model with SSD Trace Information

The correlation between $X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$, shown in Fig. 7.5, is clearly distributed around the bisecting line. According to the correlation coefficient calculated with eq. (7.6),

the correlation is found to be $\sim 86\,\%$ between the two datasets. This high correlation is also represented in the line-fit to the distribution of the datasets which has a slope of $p_0 = 0.861 \pm 0.001$.

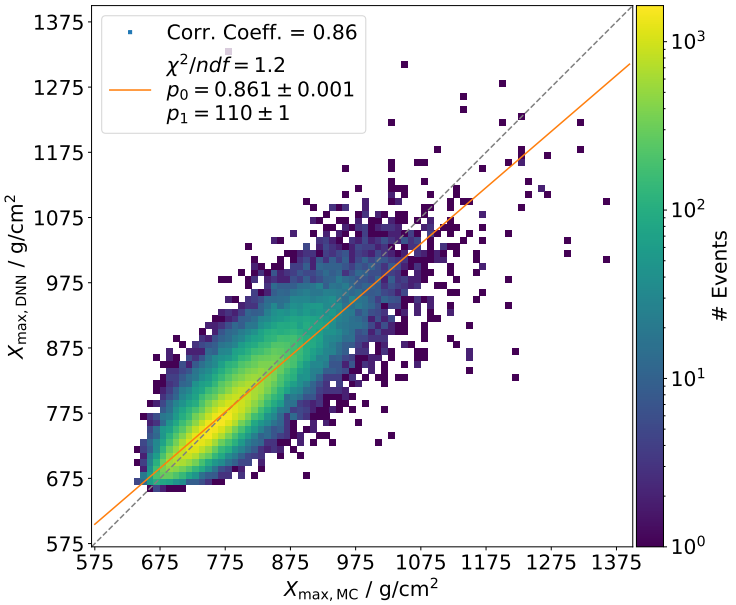


Figure 7.5: Correlation between $X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$ for $M_{\mathrm{SSD+WCD}}$. The dashed line (gray) indicates the unity line on which the data points should lie in the ideal case. The solid line (orange) is a line fit to the data points and indicates how their concentration differs from the ideal case. The correlation between the two distributions is found to be $\sim 86\,\%$.

The calculation of reconstruction bias and resolution described earlier is carried out first for each primary individually and then for the full test data set. The combined results are hereafter referred to as the *full composition* results. The calculated bias $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ is shown in Fig. 7.6 for each energy bin. At the lowest energies, $E_{\mathrm{SD,rec}} < 18.9 \log_{10}(E/\mathrm{eV})$, there is a gap in the reconstruction bias between proton and iron. This reaches a maximum of $\sim 16\,\mathrm{g/cm^2}$ at $18.5 \log_{10}(E/\mathrm{eV})$, which means that there the network over-predicts X_{max} more in the case of an iron primary than it does for protons. This effect also occurs for helium and oxygen, albeit much smaller with a maximum difference of $\sim 8\,\mathrm{g/cm^2}$. However, these composition dependent biases of the different primaries decrease and then disappear with increasing energy. As a result of this, at energy $E_{\mathrm{SD,rec}}$ above $18.9 \log_{10}(E/\mathrm{eV})$ proton and iron show a very similar behavior, as well as helium and oxygen. This causes an interesting ordering of the different primaries and a small gap in reconstruction bias between the two groups. An expected result would be for the helium and oxygen data points to lie in between proton and iron. This unexpected effect is therefore difficult to explain. With further tweaks and improvement of the network architecture in the future, this is something that could be addressed. However, due to hardware outages it was not possible to be included in this thesis.

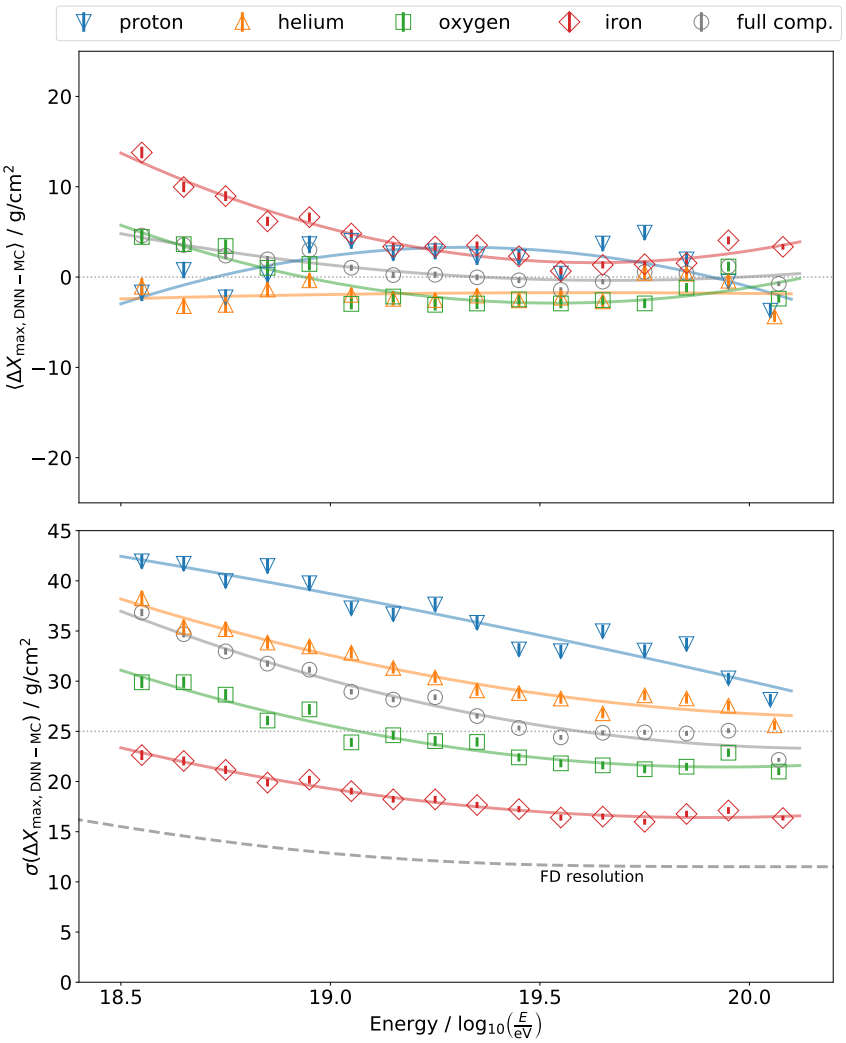


Figure 7.6: Top: Reconstruction bias $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ for proton (blue, down-triangle), helium (orange, up-triangle), oxygen (green, square), iron (red, diamond), and full composition (gray, circle) as a function of $E_{\mathrm{SD,rec}}$. Bottom: Reconstruction resolution $\sigma\,(\Delta X_{\mathrm{max,DNN\text{-}MC}})$ for proton, helium, oxygen, iron, and full composition vs. $E_{\mathrm{SD,rec}}$. In both cases the the error-bars indicate errors of the Gauss-fits and the data points are fit with 2$^{\mathrm{nd}}$ order polynomials to guide the eye.

The reconstruction resolution $\sigma\,(\Delta X_{\mathrm{max,DNN\text{-}MC}})$ can also be seen in Fig. 7.6. While it improves with increasing energies for each primary there is a constant offset between them. The reason for these differences in the resolution between the primaries is very likely a result of the network learning the form of the average underlying distribution of X_{max} in general. If the $X_{\mathrm{max,MC}}$ or $X_{\mathrm{max,DNN}}$ distribution are split by primary as given in Fig. 7.7, it can be seen that the distribution is much narrower for iron than it is for proton. As a result the neural network has a better resolution for iron than it does for protons. The overall resolution of the trained model, as shown by the full composition data points in Fig. 7.6, gets as low as $\sim 22\,\mathrm{g/cm^2}$. However, that is only the case at the highest energies. Above $19.4\log_{10}(E/\mathrm{eV})$ the resolution is well described as being a constant $\sim 25\,\mathrm{g/cm^2}$.

The resolution, for iron especially, gets relatively close to the detector resolution of the FD [78], which is indicated in Fig. 7.6.

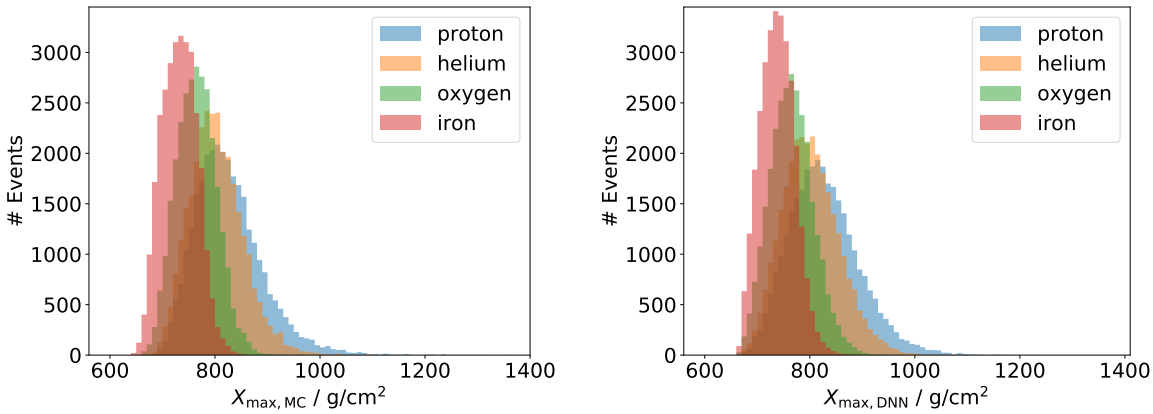


Figure 7.7: $X_{\mathrm{max,MC}}$ (left) and $X_{\mathrm{max,DNN}}$ (right) split by primary for the test data set. Lighter primaries have their shower maximum on average deeper in the atmosphere than heavier ones.

7.3.4 Evaluating the Model without SSD Trace Information

The correlation between $X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$ for the model trained without SSD trace information is shown in Fig. 7.8.

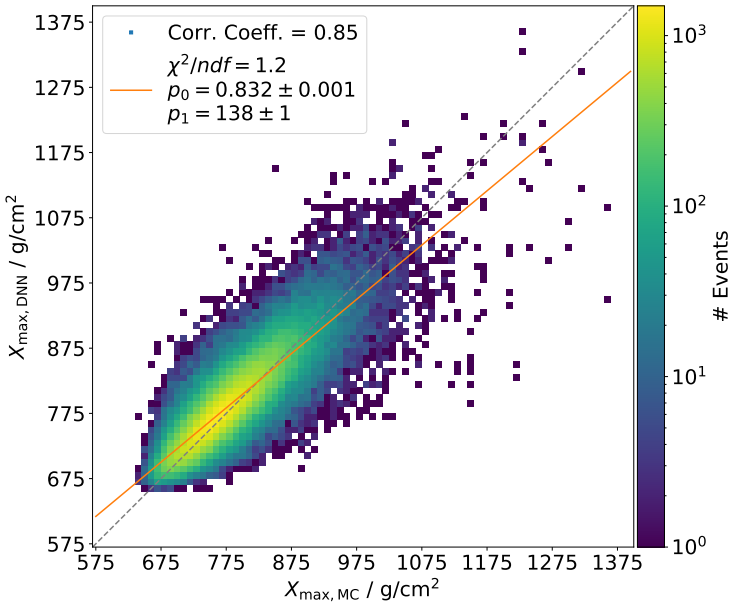


Figure 7.8: Correlation between $X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$ for M_{WCD}. The dashed (gray) and solid line (orange) again represent the unity line and a line fit to the data. The correlation between the two distributions is found to be $\sim 85\,\%$.

With a value of $\sim 85\,\%$ it is slightly lower than when $S(t)_{\mathrm{SSD}}$ is included. This is also evident from the slope of the line fit $p_0 = 0.832 \pm 0.001$, which is flatter than in the other case. While a decrease of $1.16\,\%$ in correlation is not necessarily significant, a greater impact of the missing SSD information is evident from Fig. 7.9. At energies below $18.9\log_{10}(E/\mathrm{eV})$ the gap due to composition dependent bias is larger with $\sim 25\,\mathrm{g/cm^2}$ compared to the $\sim 16\,\mathrm{g/cm^2}$ before. Also, overall the gap does not decrease with increasing energy to the extent it does with the addition of the SSD information, going only as low as $\sim 4\,\mathrm{g/cm^2}$. As a result no clear ordering of the primary dependent biases is visible.

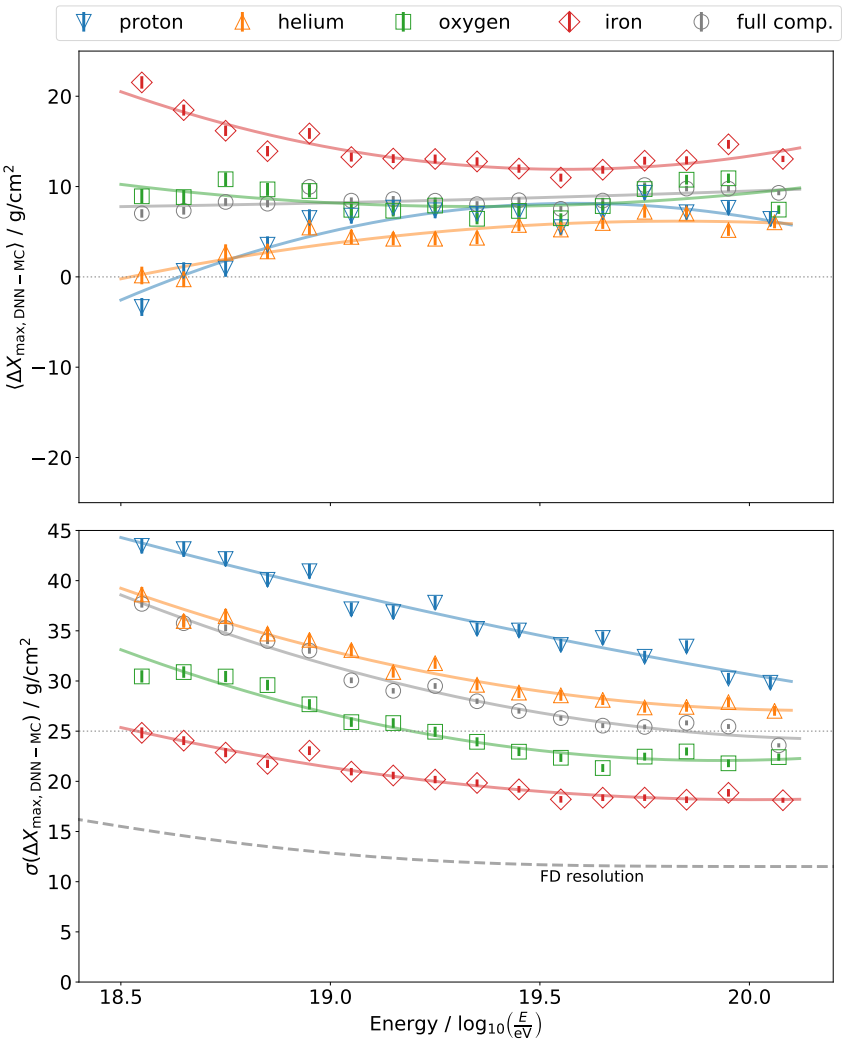


Figure 7.9: Top: Reconstruction bias $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ for M_{WCD} split by primary as a function of $E_{\mathrm{SD,rec}}$. Bottom: Reconstruction resolution $\sigma\left(\Delta X_{\mathrm{max,DNN\text{-}MC}}\right)$ split by primary vs. $E_{\mathrm{SD,rec}}$. The same symbols and color representation as in Fig. 7.6 is used.

Additionally, there is a general offset in reconstruction bias from the zero line, which means that the model generally over-predicts the X_{max}. This however, is not an issue, as a constant offset such as the one in Fig. 7.9 can be removed by correcting the $X_{\mathrm{max,DNN}}$ values by $\sim 10\,\mathrm{g/cm^2}$. The reason for this offset could be the event-weighting described in

section 7.2.1, as the network sees and advantage in "guessing" higher X_{max} values when there is an uncertainty in the result.

While the resolution $\sigma\left(\Delta X_{\mathrm{max,DNN\text{-}MC}}\right)$ of the model trained without SSD information is comparable for proton-induced showers, the resolution for the other three primaries is worse. The full-composition resolution is overall 1–2 g/cm^2 better with the SSD trace information. In the case of M_{WCD} it stays slightly above 25 g/cm^2 above $19.4\log_{10}\left(E/\mathrm{eV}\right)$.

7.4 Zenith Angle Quality Cuts and Energy-dependent Corrections

One aspect to consider for a study like the one presented here, is the varying efficiency of the SD at different zenith angles. This is especially important when working with the SSD, which has a much smaller zenith efficiency range than the WCD due to its flat shape. To account for this, potential biases in zenith can be identified and quality cuts can be applied to $X_{\mathrm{max,DNN}}$ results. In Fig. 7.10 the reconstruction bias $\langle\Delta X_{\mathrm{max,DNN\text{-}MC}}\rangle$ in relation to the zenith angle $\theta_{\mathrm{SD,rec}}$ is shown for both trained models.

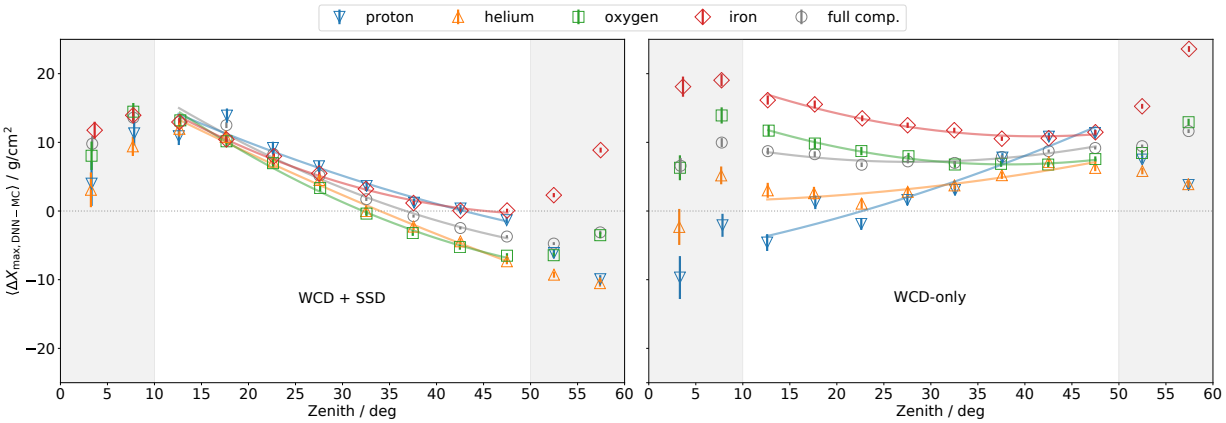


Figure 7.10: Reconstruction bias $\langle\Delta X_{\mathrm{max,DNN\text{-}MC}}\rangle$ as a function of $\theta_{\mathrm{SD,rec}}$ for all energies[1] and for both models. Left: A clear zenith dependence for the network trained with SSD information is visible with good agreement between the different primaries. Right: The model without SSD information shows no strong zenith dependence, but a strong composition dependent bias.

In the case of the model trained with SSD information, a clear zenith dependence is visible in $\langle\Delta X_{\mathrm{max,DNN\text{-}MC}}\rangle$, that is very similar for all primaries. As mentioned earlier this dependence is expected when considering the geometry of the SSD and the detector efficiency varies for different zenith angles, which leads to differences in the signal trace. This strong agreement between the different primaries is very important because as a result of this, the zenith dependence of the bias can be corrected and removed. Additionally,

[1]Simulated with E^{-1} spectrum, which is accounted for by the subsequent energy-dependent corrections.

there are strong primary dependent biases that do not follow the behavior of the other data points, at zenith angles below 10° and above 50°. The reason for this, particularly above 50°, is clear from Fig. 4.10 showing the SSD signal as a function of zenith angle, where a clear decrease in efficiency above $\theta_{\mathrm{SD,rec}} > 50°$ is seen. The behavior seen below 10°, in both $M_{\mathrm{SSD+WCD}}$ and M_{WCD}, could stem from events with X_{max} below the ground level. For vertical air showers the atmospheric depth of the ground is minimal, which means that for these the rate of showers where X_{max} is below the ground is higher. The WCD-only model has a flatter bias for the full composition, however the zenith dependence differs a lot between the simulated primaries. Because in data the primary is not available a priori, it is not possible to correct for this behavior, meaning it will bias the data reconstruction. From these results, a simple cut to reject the $\theta_{\mathrm{SD,rec}}$ ranges [0°, 10°) and (50°, 60°] is imposed. This cut has the effect of removing the events that diverge from the clear zenith dependence in the model with SSD information and it also removes the events with the biggest composition dependent biases in the WCD-only model, ensuring that the two models stay one-to-one comparable.

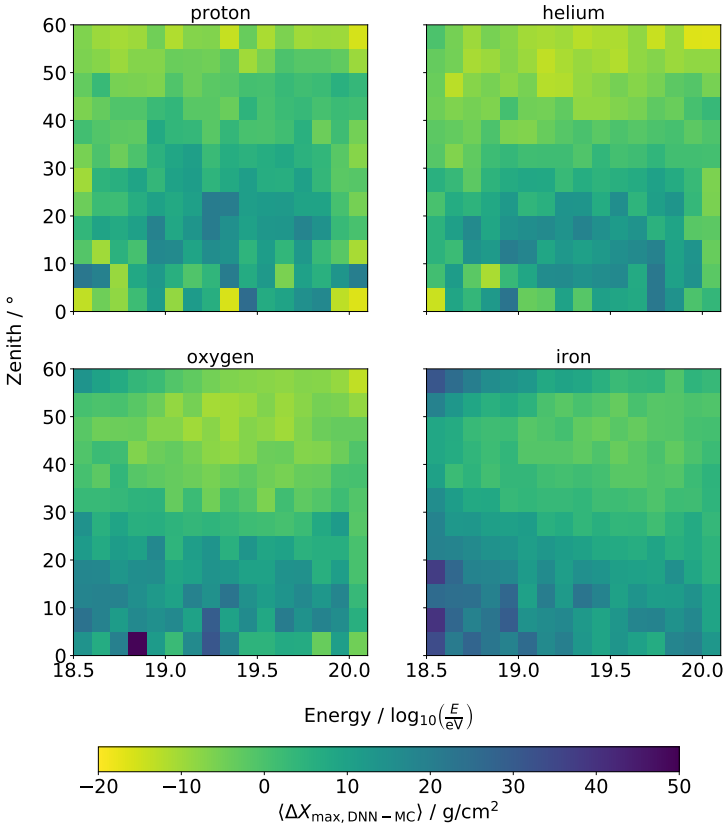


Figure 7.11: Zenith dependence of $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ for each energy bin split by primary. The color bar indicates the bias. While the overall zenith dependence is very similar between the four primaries, a lessening of the effect at the highest energies is visible, indicating an energy dependence.

When considering Fig. 7.10, the zenith dependence of $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ can be well described by a line fit to the full composition for $M_{\mathrm{SSD+WCD}}$, in the remaining zenith range

of [10°, 50°], as differences between the four simulated primaries are minimal. However, before attempting to correct for this zenith dependence, a possible primary and/or energy dependence of the effect has to be evaluated. To check for an energy or primary dependence to this zenith dependence of the bias $\langle \Delta X_{\text{max,DNN-MC}} \rangle$, the data is further split by energy and primary particle and the result is shown in Fig. 7.11. It is apparent that, while all primary particles exhibit a similar behavior, there is an energy dependence to the zenith dependence, with the zenith dependence lessening with increasing energy.

To correct for this, a two-dimensional parameterization of the energy-dependent zenith dependence is performed. As a first step, the full composition reconstruction bias $\langle \Delta X_{\text{max,DNN-MC}} \rangle$ in relation to $\theta_{\text{SD,rec}}$ is fit with a line of the form

$$\Delta X_{\text{max,DNN-MC}}(\theta_{\text{SD,rec}}) = m \cdot \theta_{\text{SD,rec}} + n \,, \tag{7.9}$$

for each $0.1 \log_{10}(E/\text{eV})$ energy bin. The values for the slope, m, and the offset, n, are shown in Fig. 7.12 for each energy bin. The behaviors of m and n as a function of energy can both be described well by 2$^{\text{nd}}$ order polynomials and are therefore fit with a function of the form

$$g(E) = p_0 \cdot E^2_{\text{SD,rec}} + p_1 \cdot E_{\text{SD,rec}} + p_2 \,, \tag{7.10}$$

with $\theta_{\text{SD,rec}}$ in units of $\log_{10}(E/\text{eV})$. The resulting fits are also visible in Fig. 7.12 together with the obtained fit parameters. With these parameters, the energy-dependent zenith correction can be applied to each event by calculating the event specific values

$$m_{\text{event}} = g_m(E^{\text{event}}_{\text{SD,rec}}) \quad \text{and} \quad n_{\text{event}} = g_n(E^{\text{event}}_{\text{SD,rec}}) \tag{7.11}$$

based on the event energy $E_{\text{SD,rec}}$. These event-specific values are then used

$$X^{\text{event}'}_{\text{max,DNN}} = X^{\text{event}}_{\text{max,DNN}} - (m_{\text{event}} \cdot \theta_{\text{SD,rec}} + n_{\text{event}}) \,. \tag{7.12}$$

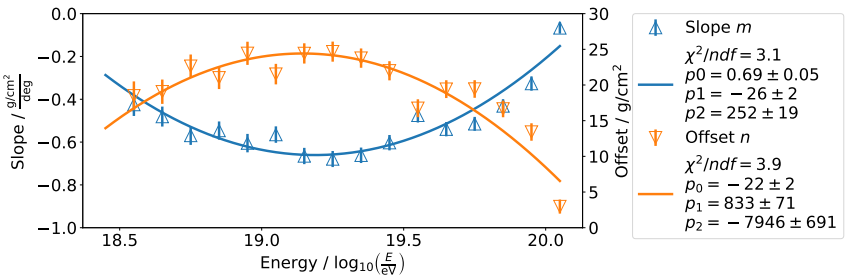


Figure 7.12: Fitted slope m (blue, up-triangle) and offset n (orange, down-triangle) obtained for each energy bin. The error-bars represent the errors of the line fits, while the solid lines show the 2$^{\text{nd}}$ order polynomial fit to the data points.

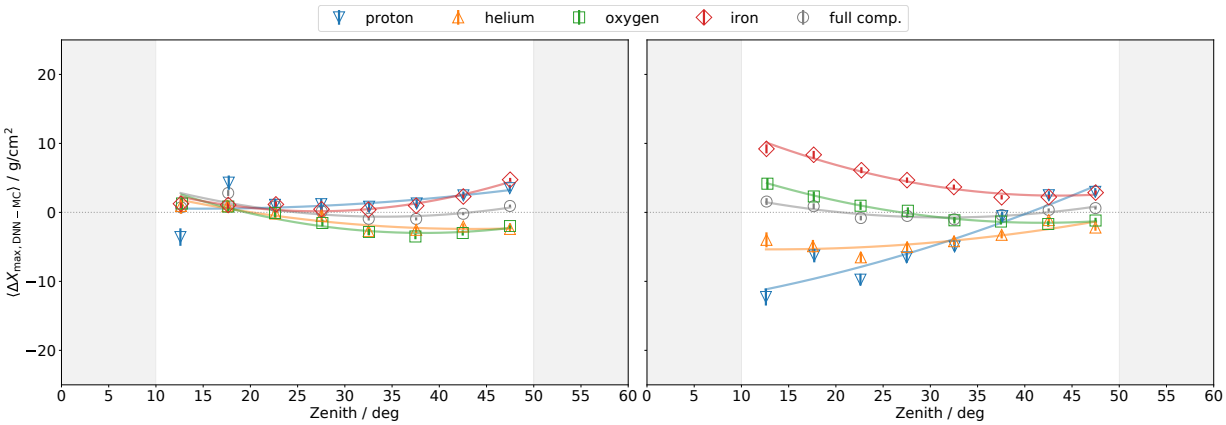


Figure 7.13: Reconstruction bias $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ as a function of $\theta_{\mathrm{SD,rec}}$ for both models after corrections. Left: The zenith dependence for the network trained with SSD information is removed for the full composition. The gray bands indicate the zenith ranges that were completely cut. Bottom: The model without SSD information now has no more offset for the full composition. The gray bands here also indicate the zenith cuts.

While this procedure was developed to be applied to the correctable behavior seen in the result of the model trained with SSD data, it can also be applied to the WCD-only model. Because the behavior of each primary is different, doing so can not improve the composition dependent bias of the individual primaries in Fig. 7.10, but it does correct the overall zenith dependence of the full composition bias to lie on the zero-line. The zenith dependence of $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ after applying these corrections is shown again for both networks in Fig. 7.13.

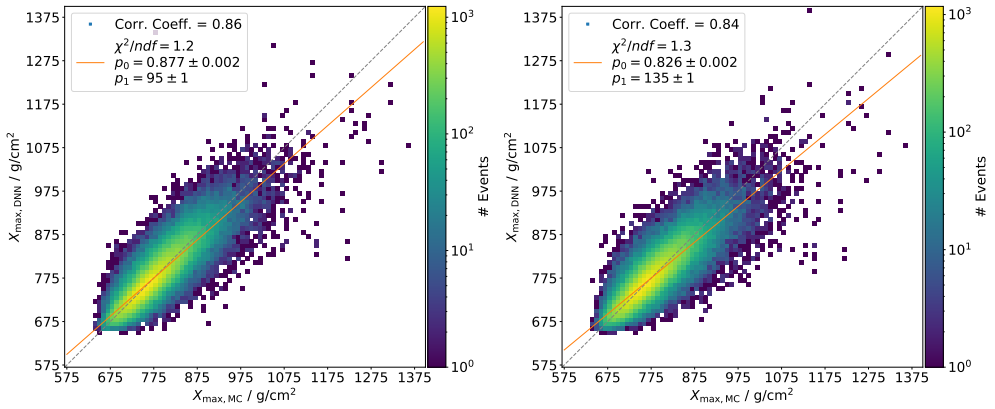


Figure 7.14: Correlation between $X_{\mathrm{max,MC}}$ and $X_{\mathrm{max,DNN}}$ for $M_{\mathrm{SSD+WCD}}$ (left) and M_{WCD} (right). The dashed line (gray) indicates the bisecting line, while the solid line (orange) is a line fit to the data points. For $M_{\mathrm{SSD+WCD}}$ the correlation stayed the same after corrections, however for M_{WCD} it has slightly decreased.

For both models, the corrected $X_{\mathrm{max,DNN}}$ values are then used to re-evaluate the previous results, i.e. the correlation between $X_{\mathrm{max,DNN}}$ and $X_{\mathrm{max,MC}}$, and each models reconstruction bias and resolution as a function of energy. The correlation after applying the cuts and

correction is shown for both models in Fig. 7.14. While the correlation coefficient stays the same for $M_{\mathrm{SSD+WCD}}$ with $\sim 86\,\%$, the slope of the line-fit improves to $p_0 = 0.877 \pm 0.002$. For M_{WCD} however, the correlation decreases to $\sim 84\,\%$, and the slope of the line-fit to $p_0 = 0.826 \pm 0.002$. These slight decreases are likely due to the removal of the high-zenith events, which are still well within the zenith-efficiency range of the WCD.

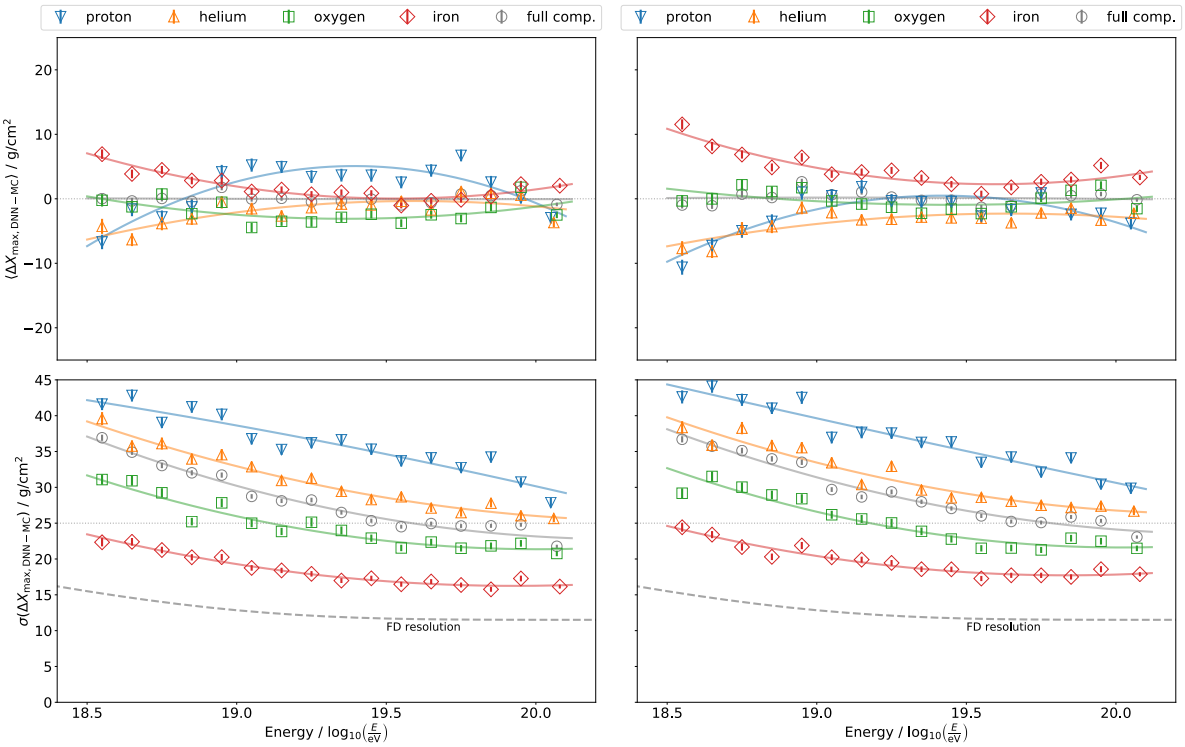


Figure 7.15: Top row: Reconstruction bias $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ for proton (blue, down-triangle), helium (orange, up-triangle), oxygen (green, square), iron (red, diamond), and full composition (gray, circle) as a function of $E_{\mathrm{SD,rec}}$ for $M_{\mathrm{SSD+WCD}}$ (left) and M_{WCD} (right) after zenith corrections. Bottom row: Reconstruction resolution $\sigma\,(\Delta X_{\mathrm{max,DNN\text{-}MC}})$ for proton, helium, oxygen, iron, and full composition vs. $E_{\mathrm{SD,rec}}$ for $M_{\mathrm{SSD+WCD}}$ (left) and M_{WCD} (right) after zenith corrections. In all cases the error-bars indicate errors of the fits.

The effect of the zenith cuts and correction on $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ and $\sigma\,(\Delta X_{\mathrm{max,DNN\text{-}MC}})$ can be seen in Fig. 7.15. For $M_{\mathrm{SSD+WCD}}$ an overall improvement of the bias is visible. The composition dependent bias below $18.9\log_{10}(E/\mathrm{eV})$ is now smaller with $\sim 13\,\mathrm{g/cm^2}$ and above $18.9\log_{10}(E/\mathrm{eV})$ the data points of the individual primaries move much closer to the zero-line making the composition dependent bias helium, oxygen and iron almost non-existent. The full composition bias is now completely flat across all energies. However the cuts and corrections also have the effect of moving the proton-only data points event further away from the rest of the primaries. Essentially, this means that with the cuts and corrections in place, since the model over-predicts proton as compared to the other primaries, it is better at predicting the particle type than it is at predicting its X_{max} in this energy range. Lastly, the overall effects on the reconstruction resolution are minimal.

For the WCD-only model, the bias offset is corrected as can be seen in Fig. 7.15. Furthermore, the gap due to composition dependent bias at the lowest energies is now slightly smaller with $\sim 22\,\mathrm{g/cm^2}$ compared to $\sim 25\,\mathrm{g/cm^2}$ before. While the gap above $18.9\log_{10}(E/\mathrm{eV})$ is now smaller than without cuts and corrections, it still does not fully disappear, only going as low as $\sim 3\,\mathrm{g/cm^2}$. Similar to the model trained with SSD information, the change in resolution is minimal.

7.5 Improvement with the Addition of SSD Data

As a last step in this analysis, the improvement in the composition sensitivity of the SD through the addition of the SSD trace information, is evaluated. Both the composition dependent biases and the full composition resolutions of the two models are compared using the $X_{\mathrm{max,DNN}}$ values with quality cuts and corrections. The overall results of this comparison can be seen in Fig. 7.16.

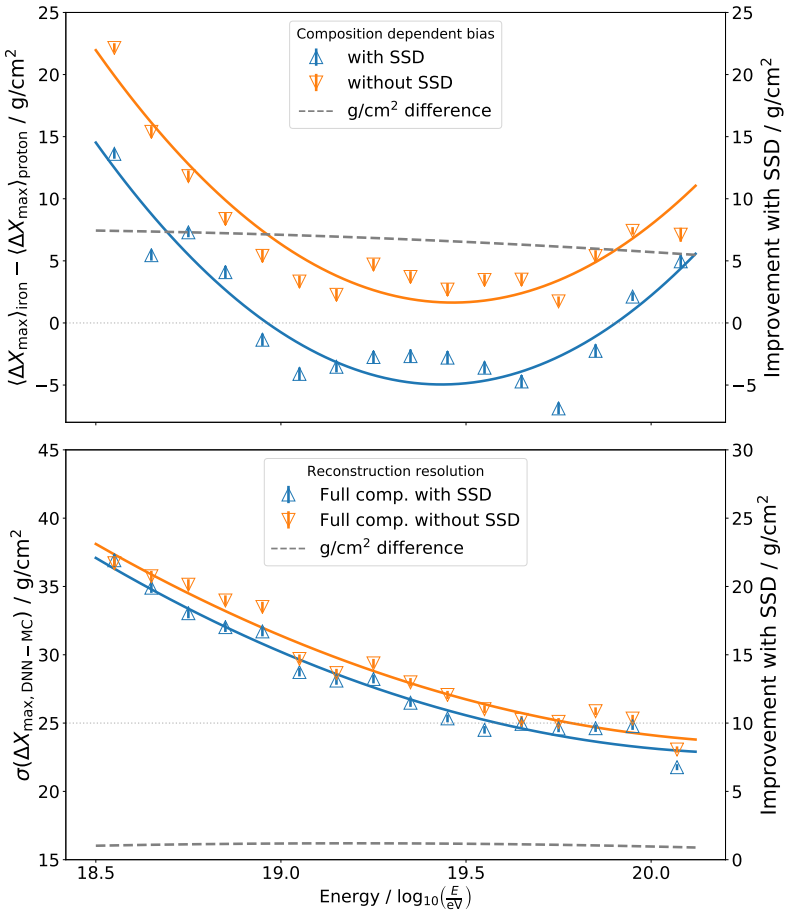


Figure 7.16: Top: Difference between $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ of iron and proton primaries for $M_{\mathrm{SSD+WCD}}$ (up-triangle, blue) and M_{WCD} (down-triangle, orange). The dashed line (gray) shows the difference between the two fits to the data points (solid lines). Bottom: The full composition distributions for both models, with the same notation as above. The dashed line (gray) represents the absolute difference between the two fit lines.

The composition dependent bias is here represented by the difference between the $\langle \Delta X_{\text{max,DNN-MC}} \rangle$ iron and $\langle \Delta X_{\text{max,DNN-MC}} \rangle$ proton data points and is fit with a 2$^{\text{nd}}$ order polynomial. As a comparison of the two models, the difference between the two fits is also shown via the dashed line and is consistently 5–8 g/cm^2 lower for $M_{\text{SSD+WCD}}$ for all energies. The same is true for the resolution of the trained models, although the effect here is less, with an improvement of only to 1.2 g/cm^2.

Another metric to use in the comparison is the so called *merit factor*. In the case of evaluating and comparing the performance of the two networks, it is used to gauge how well the predicted proton and iron X_{max} distributions can be distinguished. The merit factor, mf, is calculated as

$$mf = \frac{|\langle X_{\text{max}} \rangle_{\text{p}} - \langle X_{\text{max}} \rangle_{\text{Fe}}|}{\sqrt{\sigma(X_{\text{max}})^2_{\text{p}} + \sigma(X_{\text{max}})^2_{\text{Fe}}}} \,. \tag{7.13}$$

The mf is calculated for each energy bin separately. The means of calculating the values for $\langle X_{\text{max}} \rangle$ and $\sigma(X_{\text{max}})$ and their respective errors are taken from [79]. The merit factors calculated this way for both networks are shown in Fig. 7.17.

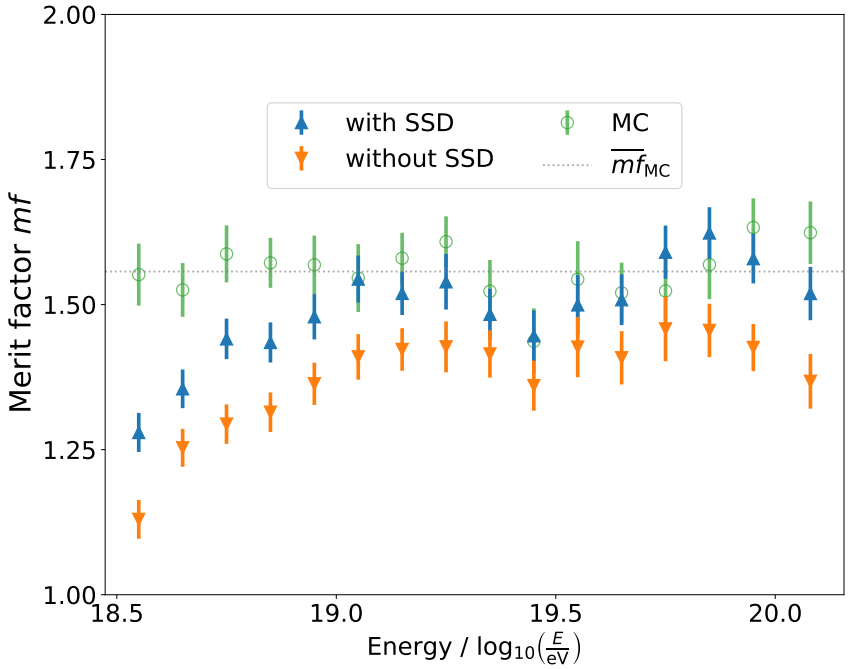


Figure 7.17: Comparison of the merit factor calculated for $M_{\text{SSD+WCD}}$ (up-triangle, blue) and M_{WCD} (down-triangle, orange) vs. energy. The average MC merit factor is indicated by the dashed line (gray). The model trained with SSD and WCD traces has a consistently better merit factor across the full energy range. The MC merit factor (circle, green) for each energy bin is also shown as a reference. Above 18.8 $\log_{10}(E/\text{eV})$ the merit factors of $M_{\text{SSD+WCD}}$ and the MC simulations always agree within uncertainties.

The merit factor of the MC distributions are also included as a reference for the theoretical maximum *mf* reachable by the two models if they perfectly predict the $X_{\mathrm{max,MC}}$ distribution. The model trained with SSD and WCD traces has a consistently higher merit factor than the WCD-only model, across the entire energy range. Additionally, above an energy of $18.9\log_{10}(E/\mathrm{eV})$, $M_{\mathrm{SSD+WCD}}$ is always in agreement with the MC merit factor within uncertainties, and at times exceeds it. Thus, the theoretically optimal distinction between proton and iron X_{max} distributions is reached.

8

Conclusion and Future Prospects

> *"It isn't all over; everything has not been invented; the human adventure is just beginning."*
>
> — Eugene Wesley "Gene" Roddenberry

In this thesis, the composition sensitivity of the AugerPrime upgrade of the Pierre Auger Observatory was studied. As a result, it was found that the resolution on the depth of shower maximum, X_{max}, and the event-by-event accuracy of its reconstruction is definitively improved through the addition of the SSD. More impressively, the measurement can be made bias-free, both with respect to mass and zenith angle, once an energy greater than $18.8 \log_{10}(E/\mathrm{eV})$ is reached.

In chapter 4, early research of data obtained from prototype detectors in the AugerPrime Engineering Array is shown. These analyses were first pursued to monitor and improve the data quality throughout the development of the upgraded electronics. An especially important step, was the verification that the updated GPS modules are able to deliver the expected detector timing accuracy of $\sim 9.4\,\mathrm{ns}$ in the 40 MHz compatibility mode. This was critical, as a good timing resolution is essential for proper reconstruction of the shower geometry. This also indicates that when the 120 MHz sampling rate is implemented, the goal timing resolution of 5–6 ns will be reached. However, these analyses also proved to be crucial to the development of the DNN. This is particularly true in the identification of parameters, besides the two different detector responses, which should be added in order to increase the prediction power of the neural network. Also, the understanding gained through these studies greatly informed on the causes and nature of different DNN behaviors observed throughout development and analysis.

Detector simulations of the fully upgraded array were then used to gauge the degree to which the AugerPrime hardware upgrade has the capability to improve composition sensitivity. The simulated data sets generated for this are presented in chapter 6. From these, a selection of input parameters was made with the goal of using only the two detector responses, $S(t)_{\mathrm{SSD}}$ and $S(t)_{\mathrm{WCD}}$, and high-level observables that can be accurately reconstructed by standard reconstruction methods. For this, the SD reconstructed energy, $E_{\mathrm{SD,rec}}$, the arrival time at the individual stations t_i, and the SD reconstructed zenith angle, $\theta_{\mathrm{SD,rec}}$ were chosen. The data was then adjusted and augmented to make it ready for processing with a DNN. This pre-processing includes down-scaling observables to smaller numbers, and also a novel method of rotating the mapping of the stations, used in order

to stabilize the network and improve the training speed. This mapping rotation results in the active stations of an event always being at a similar position within the 9×9 grid. This commonality between events means each node on average sees a similar part of the shower which results in the network learning faster. Also with this method the nodes necessary for training are reduced and thus the time needed for the network to converge decreases.

With these parameters chosen, the network architecture was developed, taking advantage of both Cartesian and hexagonal convolutions. The top half of the architecture is dedicated to comparing the two differing detector responses for each station individually via convolutions between the SSD and WCD traces. The lower half of the network then uses the features extracted from each station and combines them with the zenith, energy, and arrival time, to extract temporal and spacial features stemming from relations between the different stations. This information is then used to make a final prediction of X_{max} for each event. This network architecture was then used to train two models, one with both the SSD and WCD detector responses, $M_{\mathrm{SSD+WCD}}$, and one with only WCD data, M_{WCD}. In the network trained with both detector responses, an energy-dependent zenith dependence is found, likely due to the zenith dependent efficiency of the SSD. However, because the zenith dependence is the same for all primary types in $M_{\mathrm{SSD+WCD}}$, this zenith dependence can be corrected for. The WCD-only network shows no zenith dependence on average, but instead shows differing zenith dependencies for each primary type meaning they can not be fully corrected for. Nonetheless, the same two-dimensional parameterization method is applied to both networks and the resulting reconstruction biases $\langle \Delta X_{\mathrm{max,DNN\text{-}MC}} \rangle$ and reconstruction resolutions $\sigma\left(\Delta X_{\mathrm{max,DNN\text{-}MC}}\right)$ are compared.

After all analysis steps, it was found that the inclusion of the SSD trace information in the training of the model results in a consistently better resolution across all energies of around $1.2\,\mathrm{g/cm^2}$. Impressively, at energies below $18.9 \log_{10}(E/\mathrm{eV})$ the composition dependent bias was found to be as much as $8\,\mathrm{g/cm^2}$ smaller with $M_{\mathrm{SSD+WCD}}$ than with M_{WCD}. This holds true also for the highest energies where the bias gap remains for M_{WCD} but mostly vanishes for $M_{\mathrm{SSD+WCD}}$. In fact $M_{\mathrm{SSD+WCD}}$ here tends to over-predict the X_{max} values of protons in comparison to the other primaries, resulting in a negative bias gap. This is an indication that the network is better at identifying proton primaries than is possible with X_{max} alone. From these results, it is evident that the model trained with SSD and WCD trace information out-performs the WCD-only model in every regard.

Another deep learning effort within the Pierre Auger Collaboration, called *AixNet*, using only the WCD, has shown similar results [80] as those obtained in this thesis. The resolutions obtained with $M_{\mathrm{SSD+WCD}}$ and with AixNet are very similar. However, the composition dependent bias below $18.9 \log_{10}(E/\mathrm{eV})$ is much smaller with the $M_{\mathrm{SSD+WCD}}$ model developed in this thesis.

The merit factors for the models developed here and for AixNet are shown in Fig. 8.1. Since test set predictions made with AixNet are available within the collaboration, to

ensure one-to-one comparability, the AixNet merit factor was also calculated here for each energy bin, with the same formulas for the moments used in this thesis, which were taken from [79]. While the underlying detector simulations are not the same, the differences are primarily in the addition of the SSD and both simulations sets use the same MC library and hadronic interaction model (EPOS-LHC) and are therefore directly comparable. From this comparison it seems that the results of both WCD-only models are similar below $19.2\log_{10}(E/\mathrm{eV})$, with AixNet eventually performing better at higher energies. From Fig. 8.1 it is also clearly evident that the additional information contained in the SSD traces results in a better composition sensitivity at all energies.

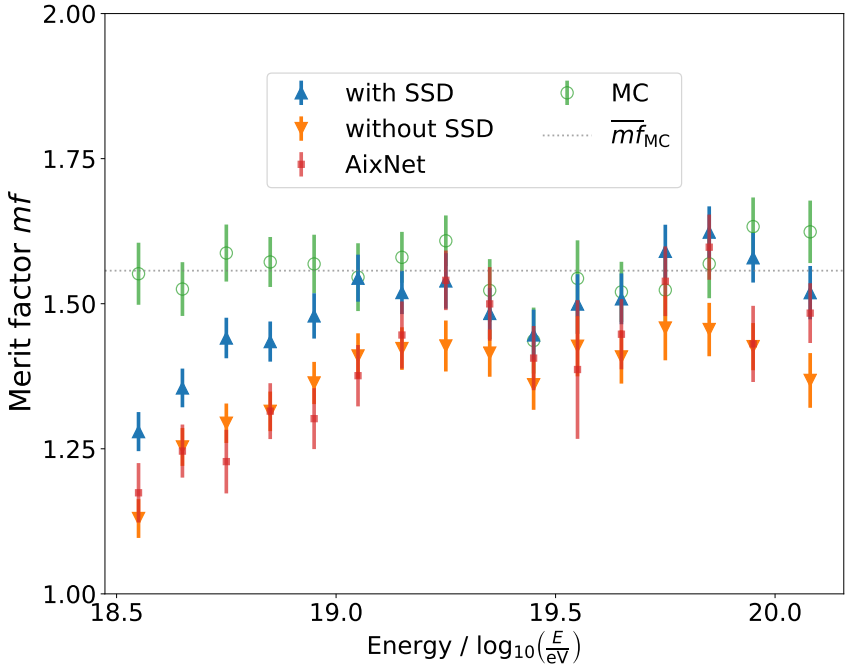


Figure 8.1: Comparison of the merit factor calculated for $M_{\mathrm{SSD+WCD}}$ (up-triangle, blue), M_{WCD} (down-triangle, orange), and AixNet (square, red) vs. energy. The average MC merit factor is indicated by the dashed line (gray). It is clear that $M_{\mathrm{SSD+WCD}}$ overall has the best proton-iron separation, due to the addition of the SSD trace information.

There is also room to improve as there is still a number of possible tweaks to the pre-processing and network architecture that could improve performance, but could not be carried out in the time frame of this thesis. In pre-processing, further augmentation of the input data is possible. In its current form, both the signal traces of the WCD and the SSD, as well as the arrival times, can reach very large values, which are not ideal for processing by neural networks. Methods to scale these three parameters down should be implemented in future studies of this kind and are currently being pursued within the collaboration.

Regarding the network architecture, the bi-directional *long short-term memory* (LSTM) layers used for AixNet, likely could further improve the performance of $M_{\mathrm{SSD+WCD}}$. This

is because they are well-suited to time series data, as they have the ability to form better connections between past and current time steps than the method used here and therefore could be used to replace the 3D Cartesian convolutions in the top half of the network architecture. Even without this change, it is expected that further tweaks in the network architecture would decrease the remaining composition dependent biases, and, if desired, the over-prediction of X_{max} for proton primaries above $18.9\log_{10}(E/\mathrm{eV})$. One prominent means of increasing performance would be to further improve the detector simulations obtained with $\overline{\mathrm{Off}}\underline{\mathrm{line}}$, a work which is currently underway. As mentioned in chapter 6, these are still in preliminary form and there is still work needed to correct the implementation of the trigger simulations. Once this is completed, the studies presented here could potentially be extended to energies below $18.5\log_{10}(E/\mathrm{eV})$.

Despite these opportunities for improvement, the merit factor results for $M_{\mathrm{SSD+WCD}}$ have shown that the theoretically optimal distinction between proton and iron X_{max} distributions is already reached above $18.9\log_{10}(E/\mathrm{eV})$. Because X_{max} is inherently random to some degree, the networks are trying to predict a random variable. This likely is a cause for the seemingly hard resolution limit of $25\,\mathrm{g/cm^2}$ seen both here and in AixNet. This means it is likely that further distinction between masses would only be possible with a transition from a regression task, where the network learns to predict the underlying distribution of X_{max}, to a classification task that directly predicts the mass or particle type. With all of this in hand, it is clear that the studies presented in this thesis show that the goal of achieving high-accuracy composition measurements on an event-by-event basis is possible with AugerPrime.

A

EA Analysis: Additional Plots

A.1 SD-433 Reconstruction and HV Stability

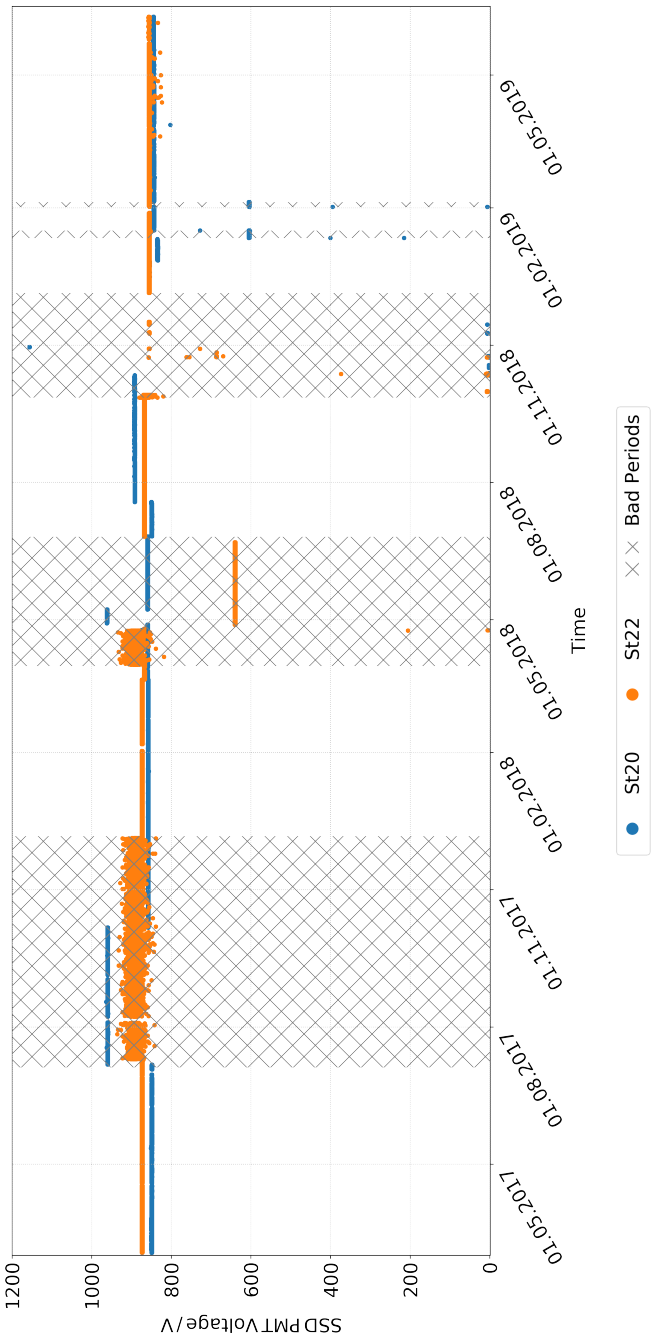


Figure A.1: HV of EA stations 20 and 22 over the full analysis period.

A.2 Timing Analysis without Shower Direction Correction

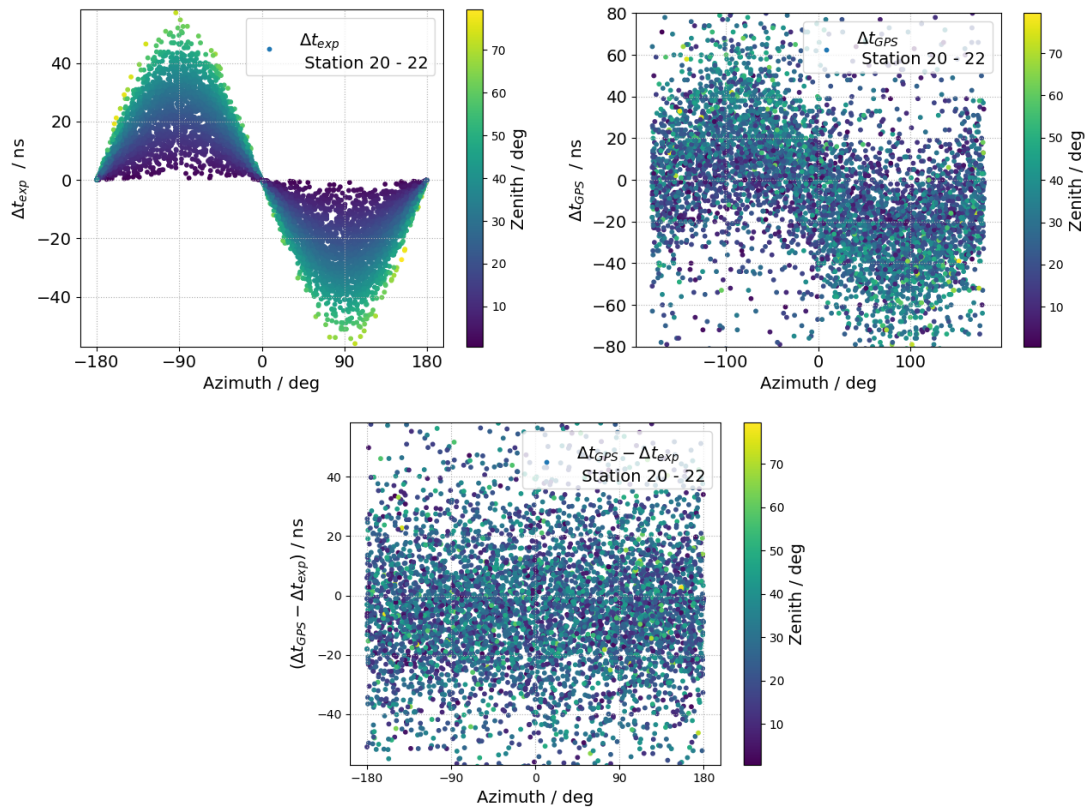


Figure A.2: Top left: Expected timing differences between station 20 and 22, based on SD_{433} reconstruction depending on azimuth angle. Top right: Timing difference based on GPS differences. Bottom: Subtraction of distributions of expected timing and GPS timing, i.e. Δt_{corr}.

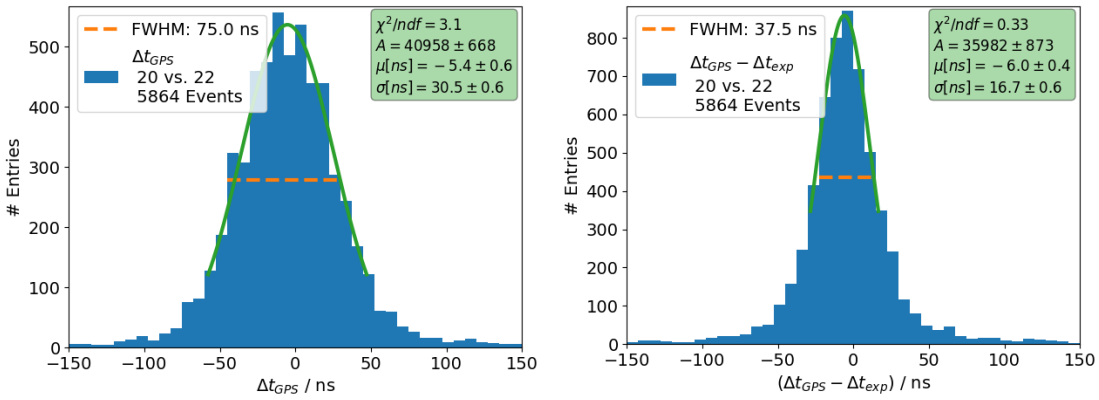


Figure A.3: Left: Distribution of Δt_{GPS} for stations 20 and 22 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 20 and 22 with Gauss-fit.

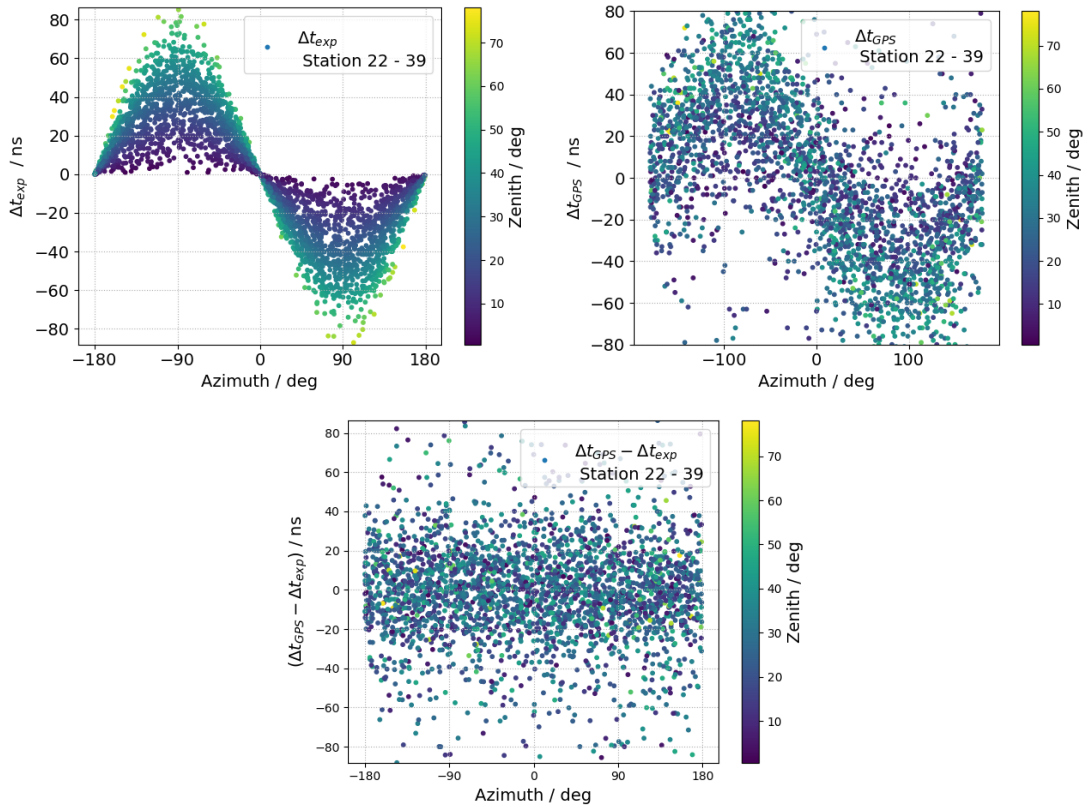


Figure A.4: Top left: Expected timing differences between station 22 and 39, based on SD_{433} reconstruction depending on azimuth angle. Top right: Timing difference based on GPS differences. Bottom: Subtraction of distributions of expected timing and GPS timing, i.e. Δt_{corr}.

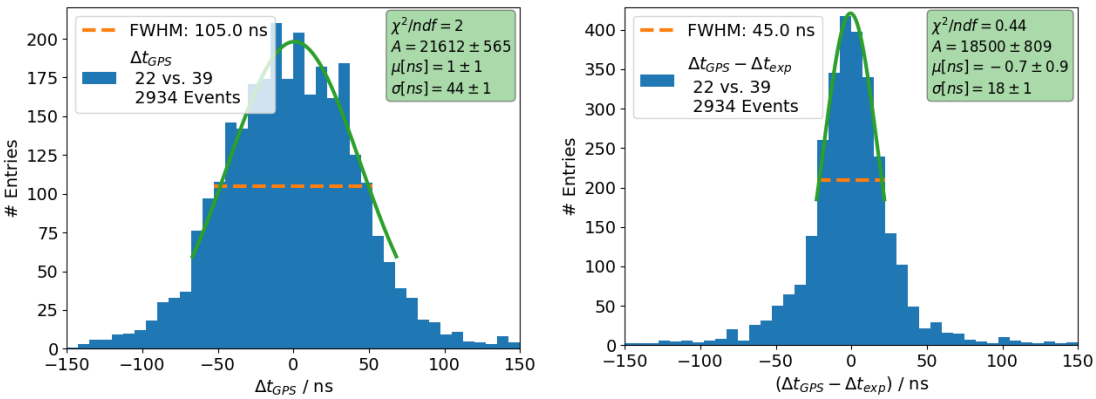


Figure A.5: Left: Distribution of Δt_{GPS} for stations 22 and 39 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 22 and 39 with Gauss-fit.

A.3 Timing Analysis during Full-Bandwidth Trigger Period

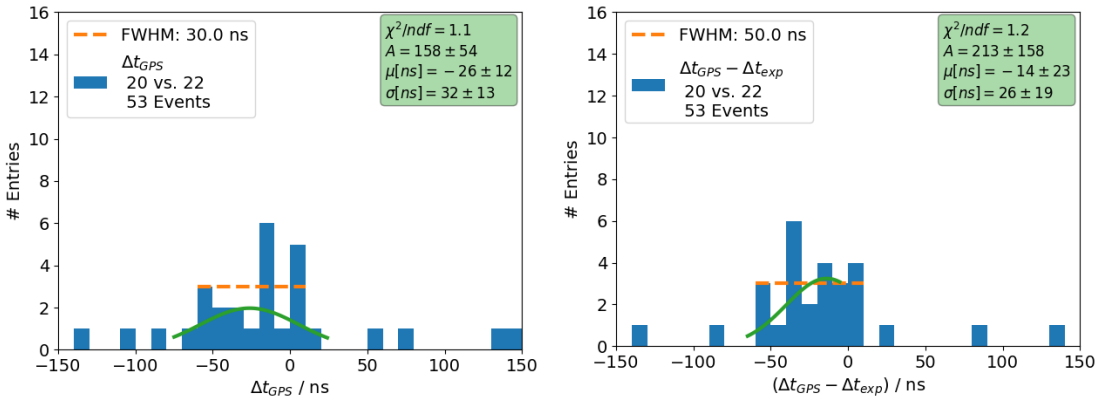


Figure A.6: Left: Distribution of Δt_{GPS} for stations 20 and 22 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 20 and 22 with Gauss-fit.

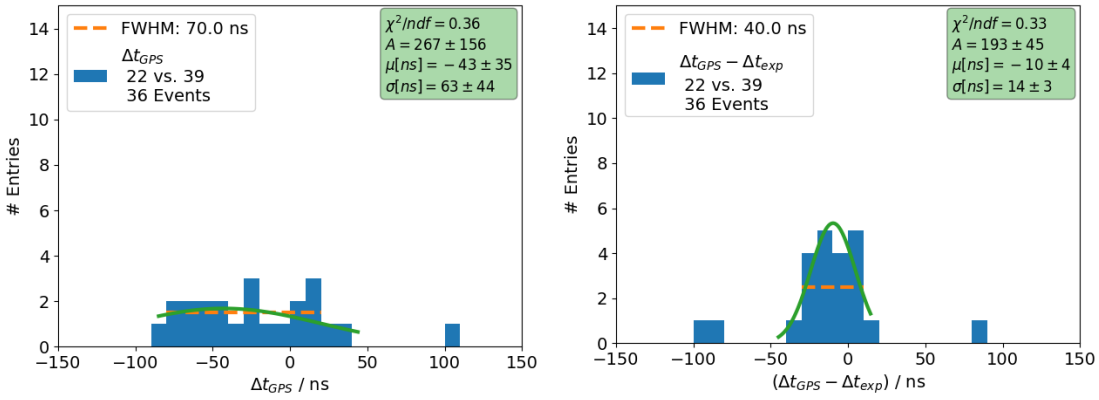


Figure A.7: Left: Distribution of Δt_{GPS} for stations 22 and 39 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 22 and 39 with Gauss-fit.

A.4 Timing Analysis after switch of GPS module

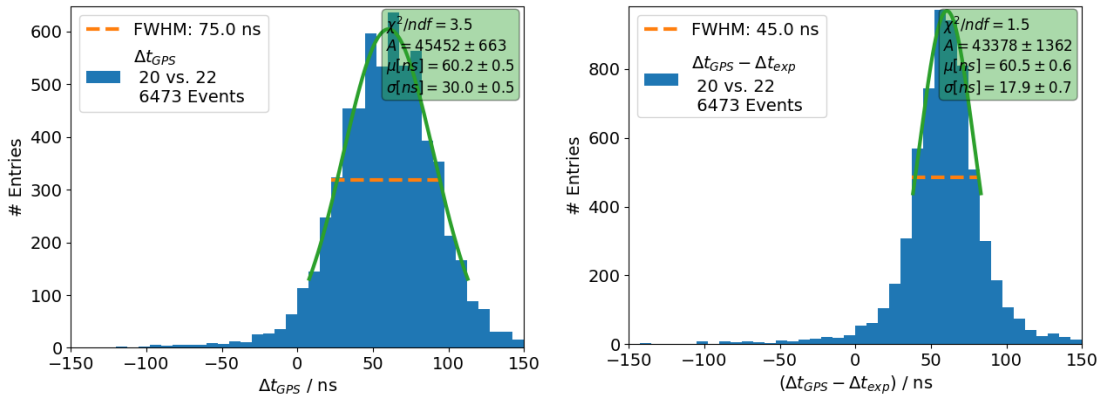


Figure A.8: Left: Distribution of Δt_{GPS} for stations 20 and 22 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 20 and 22 with Gauss-fit.

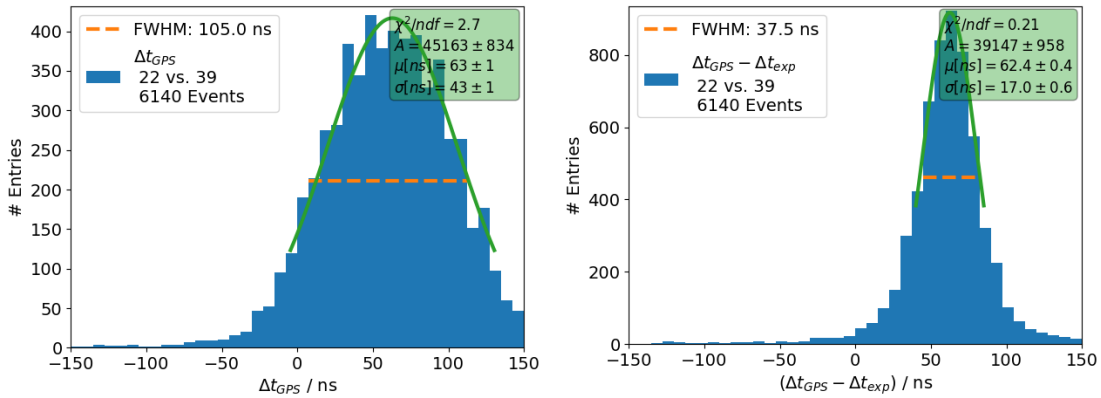


Figure A.9: Left: Distribution of Δt_{GPS} for stations 22 and 39 with Gauss-fit. Right: Distribution of Δt_{corr} for stations 22 and 39 with Gauss-fit.

A.5 MIP Peak Stability vs. Time/Temperature

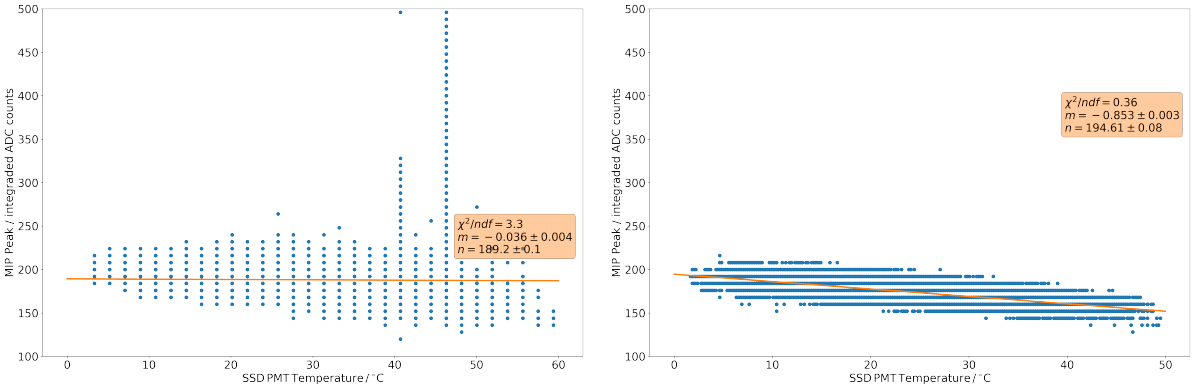


Figure A.10: MIP-temperature correlations for station 22 with UUB V1 (left) and UUB V2 (right). The refined temperature calculation is visible on the right.

A.6 Trace Baseline Stability

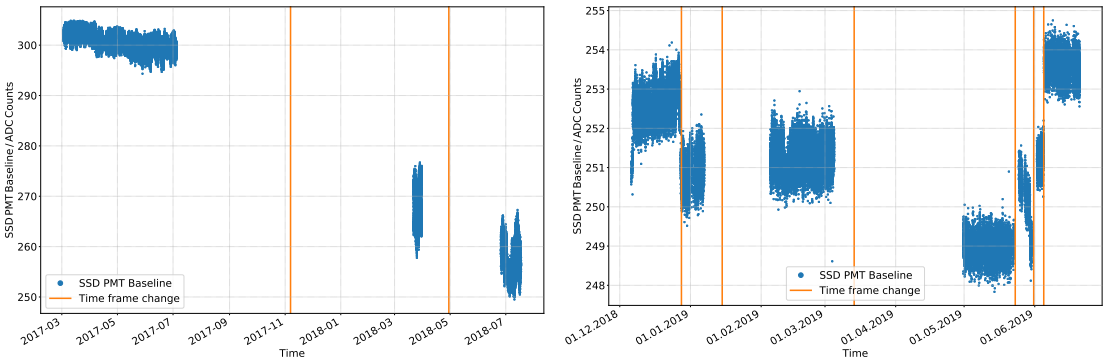


Figure A.11: Baseline vs. time for station 22 with UUB V1 (left) and UUB V2 (right). The UUB V2 shows more different popultions, due to more baseline changes.

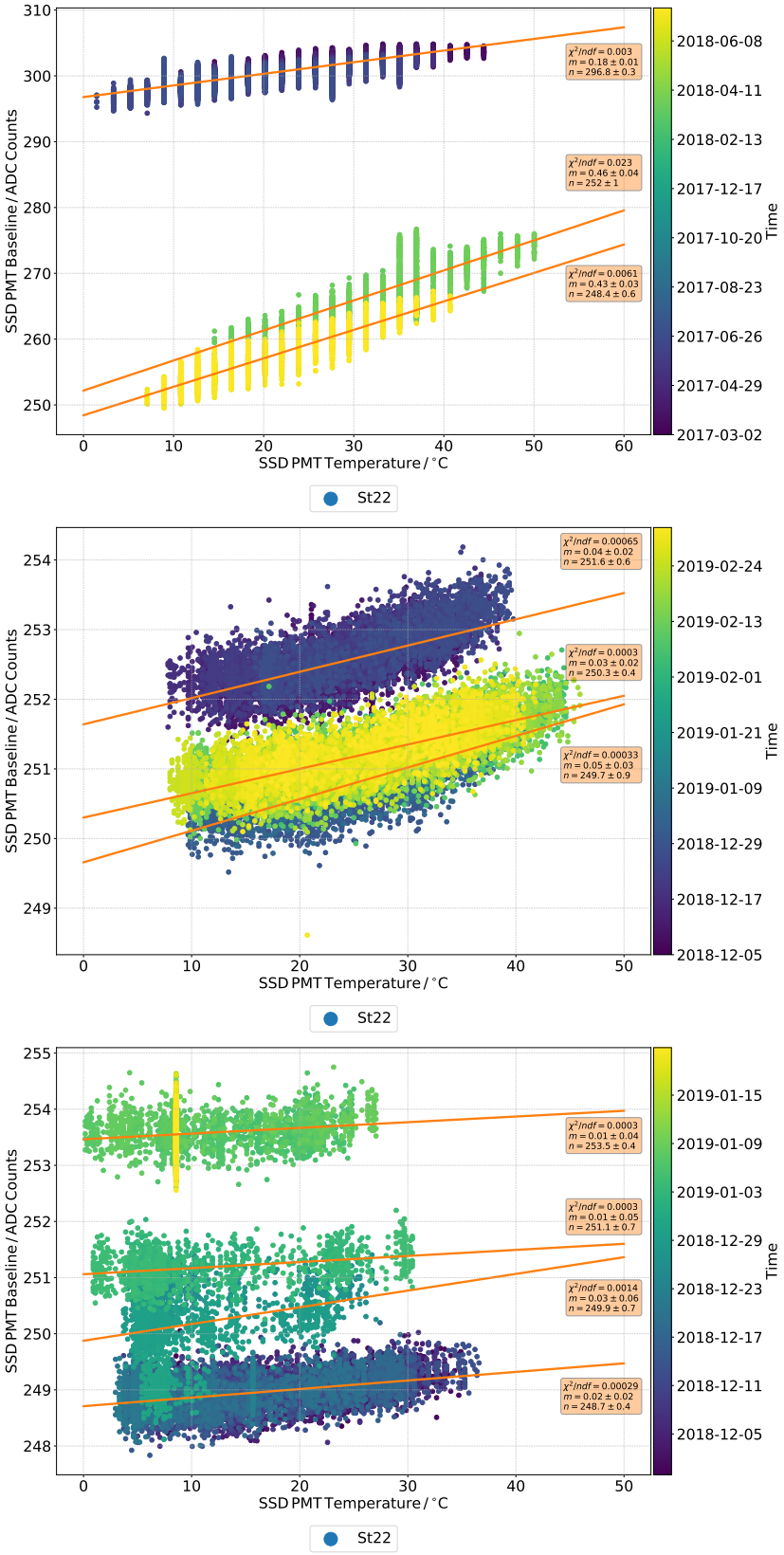
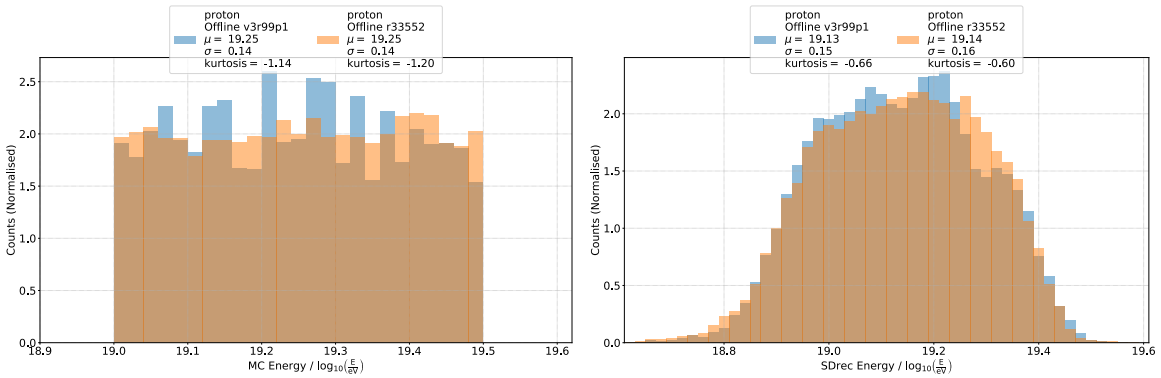


Figure A.12: Baseline-temperature correlations for station 22 with UUB V1 (left) and UUB V2 (right). The UUB V2 shows more populations, due to the numerous baseline changes.

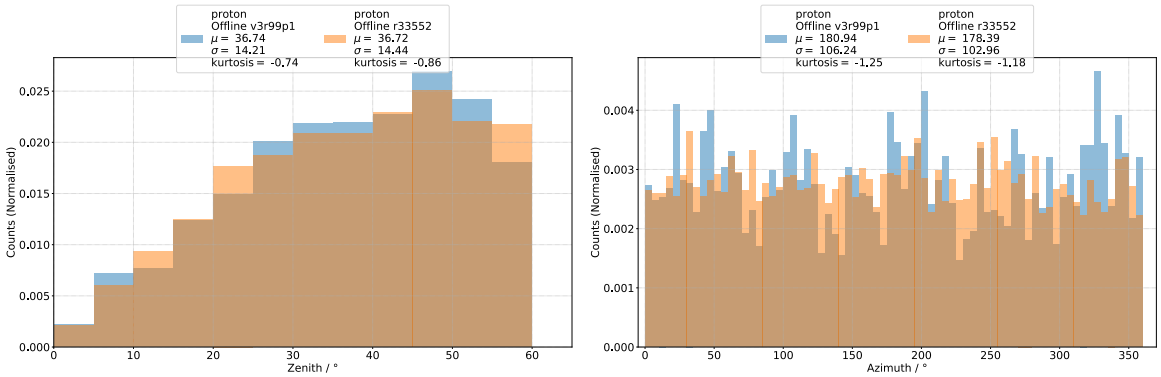
B

Comparison of two AugerPrime Simulation Sets

In an earlier version of the research presented in this thesis, a different set of simulations was used. The earlier set of simulations was made for both iron and proton primaries locally in Wuppertal (here also referred to as BUW) with the most recent version of Offline available at the time, with the tag `v3r99p1`. However, higher statistics and additional primaries were needed to train the neural networks presented here. The Collaborations MC task was more well suited to produce such an AugerPrime simulation library.



(a) E_{MC} **(b)** $E_{\mathrm{SD,rec}}$



(c) $\theta_{\mathrm{SD,rec}}$ **(d)** $\phi_{\mathrm{SD,rec}}$

Figure B.1: Distribution of event-level parameters E_{MC}, $E_{\mathrm{SD,rec}}$, $\theta_{\mathrm{SD,rec}}$, and $\phi_{\mathrm{SD,rec}}$.

There had been a lot of changes to the AugerPrime simulation code since making the first small production in Wuppertal and updates in the trigger simulation were in progress. As a result a study was needed to decide between making a larger scale simulation library with the `v3r99p1` version of Offline or a newer one. The newer version to compare the

Wuppertal simulations to was chosen to be the revision `r33552`, which contained a lot of bug fixes in regards to the GEANT4 modules.

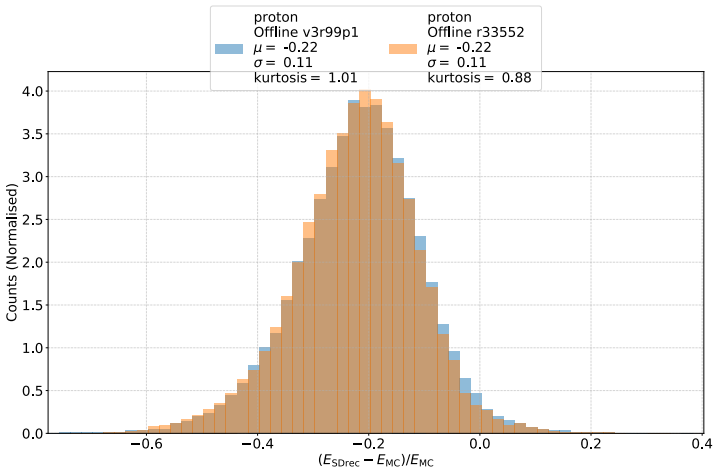


Figure B.2: Relative $E_{\mathrm{SD,rec}}$ reconstruction error.

Two small simulations set were subsequently compared, one with $\overline{\text{Off}}\underline{\text{line}}$ `v3r99p1` and on with $\overline{\text{Off}}\underline{\text{line}}$ `r33552`, both using proton as a primary and EPOS-LHC as the hadronic interaction model. The `v3r99p1` simulations were made with CORSIKA simulations also made in Wuppertal, while the revision `r33552` simulations are made with the Napoli/Praha library. All simulation are in the energy range $E_{\mathrm{MC}} \in [19.0 \log_{10}(E/\mathrm{eV}), 19.5 \log_{10}(E/\mathrm{eV})]$. As can be seen from Fig. B.1, the most important event-level distributions, i.e. E_{MC}, $E_{\mathrm{SD,rec}}$, $\theta_{\mathrm{SD,rec}}$, and $\phi_{\mathrm{SD,rec}}$ all look comparable. Both simulations sets where produced with and energy spectrum of E^{-1}, but results look better for `r33552`. Similarly both simulation sets were produced flat in azimuth, but shown slightly better performance for `r33552`. This is however, very likely due to the underlying CORSIKA simulations and not necessarily an effect of differences in the two $\overline{\text{Off}}\underline{\text{line}}$ versions.

An important aspect is that the relative error on $E_{\mathrm{SD,rec}}$, shown in Fig. B.2, is of the same quality for both versions of $\overline{\text{Off}}\underline{\text{line}}$, as this is an indicator that the SD energy reconstruction is working properly. At the time this study was carried out, the network was still trained with all stations in an event, i.e. all stations with non-zero total signal, and not just candidate stations. The average total signal in a station over all events is shown in Fig. B.3. The station color indicates the strength of the signal, while the size of the circle is scaled by frequency of "appearance" of that station in the event set. This is done separately for SSD and WCD in order to detect potential differences. While the maximum average total signal in the WCDs is comparable, the maximum average total signal in the SSD is higher in the `r33552` simulations. This is likely explainable by the different $\overline{\text{Off}}\underline{\text{line}}$ versions, due to a change in the MIP definition after `v3r99p1`, changing the magnitude of the signals. Overall, the number of stations appearing in an event looks to be higher in the `v3r99p1` set. This was especially unexpected in the WCD and could

indicate differences in the trigger simulations.

To investigate this further the station counts in the SSD than in the WCD are studied in Fig. B.4. In both cases, the station count is higher for `v3r99p1`. As an additional cross-check a single helium event from the Napoli/Praha library is simulated with both Offline versions. The station count for this event also differs between the two Offline versions, an indication that the different underlying CORSIKA simulations in the proton simulations is not the reason for the differing station counts.

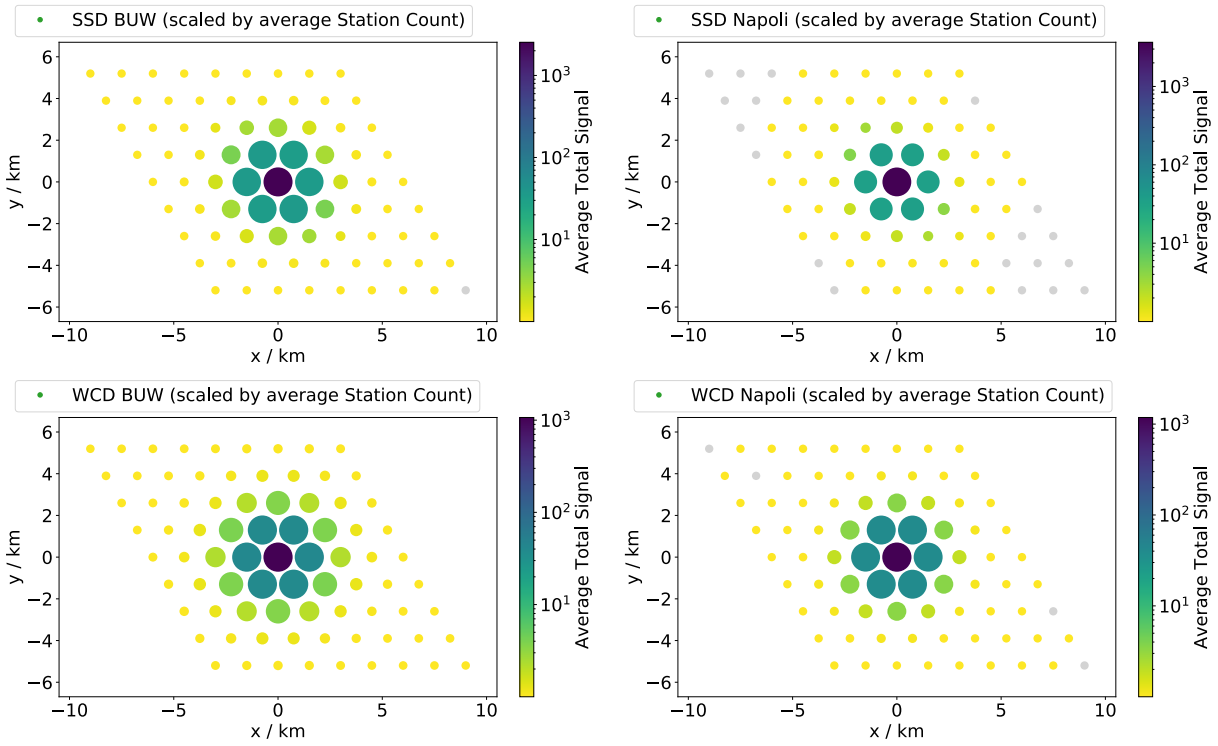
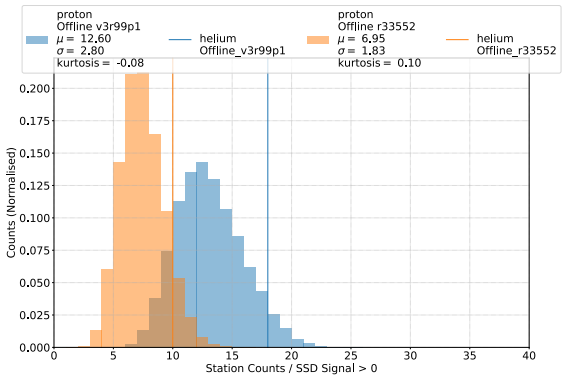


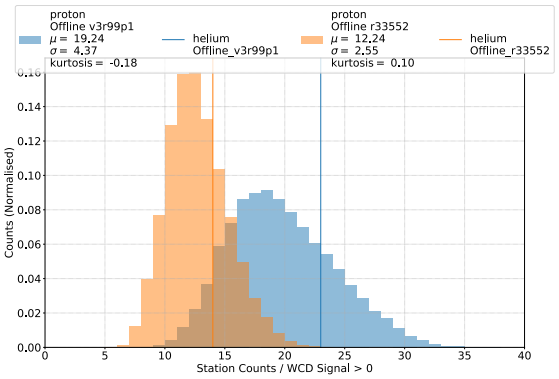
Figure B.3: Each stations average total signal over all events is calculated and indicated by the color scale. The size of a station represents its average frequency of appearance, a measure of how often a station participates in events.

The proton `v3r99p1` simulations are also compared to a small set of helium simulations from the Napoli/Praha library processed with `v3r99p1` in Wuppertal. The results of this are shown in Fig. B.4 as well and are comparable, with the small offset likely an effect due to the difference in primary particle.

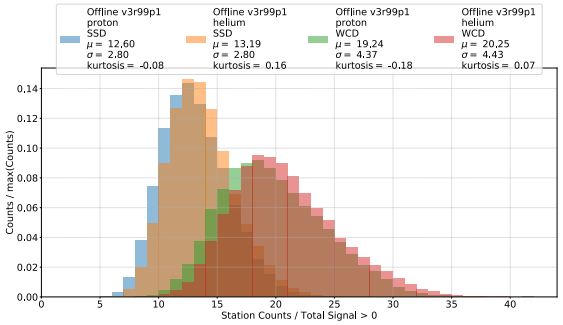
As a last cross-check the total signals in both the SSD and the WCD are studied, focusing on very low signals, see Fig. B.5. Overall, there are more stations with very low signal in both SSD and WCD for $\overline{\text{Off}}\underline{\text{line}}$ `v3r99p1`, which is probably a reason for the higher station counts. As a result of the studies presented here, Offline revision `r33552` was chosen to make a first AugerPrime simulation library and this library is the one used for the models trained and presented in this thesis. This decision was made, because the results from `r33552` are more comparable with what SD simulations of non-upgraded stations look like in regards to the WCD.



(a) SSD Counts

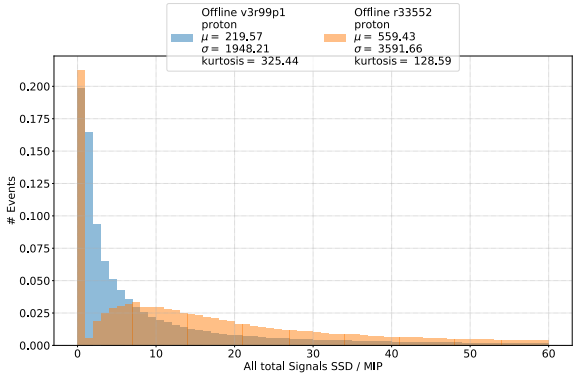


(b) WCD Counts

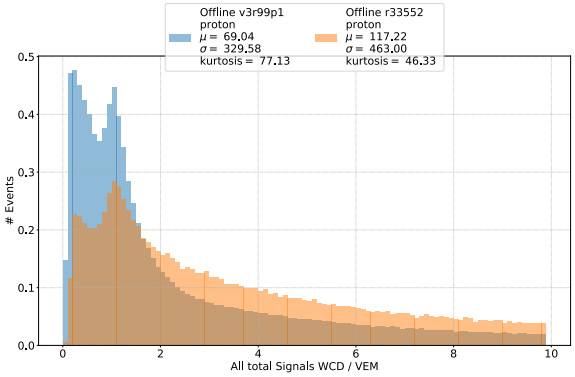


(c) BUW Counts

Figure B.4: The count of non-zero stations of all events. The histograms are normalized for comparison. The SSD and WCD counts are studied separately and differ, as a non-zero WCD station can potentially not have a signal in the SSD. The vertical lines in the top row show the respective station counts of the helium event simulated from the Napoli/Praha library with both Offline versions. The bottom plot compared proton and helium events with `v3r99p1` but different underlying CORSIKA simulations.



(a) SSD Counts



(b) WCD Counts

Figure B.5: Very low total signals for SSD (<60) and WCD (<10) compared between the two Offline versions.

C

selectEvents File

```
T4Trigger 2          # required T4 flag (0: no T4,
#                   1: FD-triggered,
#                   2: SD-triggered,
#                   3: FD- and SD-triggered (Golden hybrids))
T5Trigger 2          # required T5 flag (0: any T5,
#                   1: 5T5 posterior,
#                   2: 6T5 prior,
#                   3: T5Has)
minZenithSD 0.       # minimum zenith angle [deg]
maxZenithSD 60.      # maximum zenith angle [deg]
minRecLevel 3        # see SdRecLevel.h
minLgEnergySD 18.5   # minimum reconstructed lg(energy/[eV])
maxLgEnergySD 99.0   # maximum reconstructed lg(energy/[eV])
!lightning           # (bool) select events  that have
at least one lightning station
```

References

[1] H. Becquerel. 'On the rays emitted by phosphorescence'. In: *Compt. Rend. Hebd. Seances Acad. Sci.* 122.8 (1896), pp. 420–421.

[2] H. Geitel. 'Über die Elektrizitätszerstreuung in abgeschlossen Luftmengen'. In: *Physikalische Zeitschrift* 2 (1900), pp. 116–119.

[3] J. Elster and H. Geitel. 'Weitere Versuche über die Elektrizitätszerstreuung in abgeschlossen Luftmengen'. In: *Physikalische Zeitschrift* 2 (1901), pp. 560–563.

[4] C. T. Wilson. 'On the leakage of electricity through dust-free air'. In: *Proc. Cambridge Philos. Soc.* Vol. 11. 1900, p. 32.

[5] T. Wulf. 'Über den Ursprung der in der Atmosphäre vorhandenen γ-Strahlung'. In: *Physikalische Zeitschrift* 10 (1909), pp. 997–1003.

[6] T. Wulf. 'Beobachtungen über die Strahlung hoher Durchdringungsfähigkeit auf dem Eiffelturm'. In: *Physikalische Zeitschrift* 11 (1910), pp. 811–813.

[7] V. F. Hess. 'Über Beobachtungen der durchdringenden Strahlung bei sieben Freiballonfahrten'. In: *Phys. Z.* 13 (1912), pp. 1084–1091.

[8] P.A. Zyla et al. (Particle Data Group). 'Review of Particle Physics'. In: *Progress of Theoretical and Experimental Physics* 2020.8 (Aug. 2020). 083C01. DOI: 10.1093/ptep/ptaa104.

[9] E. Fermi. 'On the Origin of the Cosmic Radiation'. In: *Phys. Rev.* 75 (8 Apr. 1949), pp. 1169–1174. DOI: 10.1103/PhysRev.75.1169.

[10] G. M. Webb. 'First order and second order Fermi acceleration of energetic charged particles by shock waves'. In: *The Astrophysical Journal* 270 (1983), pp. 319–338. DOI: 10.1086/161125.

[11] C. Grupen. *Astroparticle Physics*. Springer, 2005. ISBN: 9783540253129. DOI: 10.1007/978-3-030-27339-2.

[12] A. Aab et al. 'Depth of maximum of air-shower profiles at the Pierre Auger Observatory. I. Measurements at energies above $10^{17.8}$ eV'. In: *Phys. Rev. D* 90 (12 Dec. 2014), p. 122005. DOI: 10.1103/PhysRevD.90.122005.

[13] A. Aab et al. 'Depth of maximum of air-shower profiles at the Pierre Auger Observatory. II. Composition implications'. In: *Phys. Rev. D* 90 (12 Dec. 2014), p. 122006. DOI: 10.1103/PhysRevD.90.122006.

[14] J. Bellido on behalf of the Pierre Auger Collaboration. 'Depth of maximum of air-shower profiles at the Pierre Auger Observatory: Measurements above $10^{17.2}$ eV and Composition Implications'. In: *Proceedings of 35th International Cosmic Ray Conference — PoS(ICRC2017)*. Vol. 301. 2017, p. 506. DOI: 10.22323/1.301.0506.

[15] A. Aab et al. 'Measurement of the cosmic-ray energy spectrum above 2.5×10^{18} eV using the Pierre Auger Observatory'. In: *Phys. Rev. D* 102 (6 Sept. 2020), p. 062005. DOI: 10.1103/PhysRevD.102.062005.

[16] A. Aab et al. 'Features of the Energy Spectrum of Cosmic Rays above 2.5×10^{18} eV Using the Pierre Auger Observatory'. In: *Phys. Rev. Lett.* 125 (12 Sept. 2020), p. 121106. DOI: 10.1103/PhysRevLett.125.121106.

[17] A. Aab et al. 'Combined fit of spectrum and composition data as measured by the Pierre Auger Observatory'. In: *JCAP* 04 (2017). [Erratum: JCAP 03, E02 (2018)], p. 038. DOI: 10.1088/1475-7516/2017/04/038.

[18] K. Greisen. 'End to the Cosmic-Ray Spectrum?' In: *Phys. Rev. Lett.* 16 (17 Apr. 1966), pp. 748–750. DOI: 10.1103/PhysRevLett.16.748.

[19] G. T. Zatsepin and V. A. Kuzmin. 'Upper limit of the spectrum of cosmic rays'. In: *JETP Lett.* 4 (1966), pp. 78–80.

[20] M. G. Aartsen et al. 'Differential limit on the extremely-high-energy cosmic neutrino flux in the presence of astrophysical background from nine years of IceCube data'. In: *Phys. Rev. D* 98 (6 Sept. 2018), p. 062003. DOI: 10.1103/PhysRevD.98.062003.

[21] P. Papenbreer. 'Search for Ultra-High-Energy Photons with the Pierre Auger Observatory'. PhD thesis. Wuppertal U., 2020. DOI: 10.25926/h3hs-t241.

[22] J. Rautenberg on behalf of the Pierre Auger Collaboration. 'Limits on ultra-high energy photons with the Pierre Auger Observatory'. In: *PoS* ICRC2019 (2021), p. 398. DOI: 10.22323/1.358.0398.

[23] T. Gaisser and A. M. Hillas. 'Reliability of the method of constant intensity cuts for reconstructing the average development of vertical showers'. In: *International Cosmic Ray Conference*. Vol. 8. 1977, pp. 353–357.

[24] K. Kamata and J. Nishimura. 'The lateral and the angular structure functions of electron showers'. In: *Progress of Theoretical Physics Supplement* 6 (1958), pp. 93–155. DOI: 10.1143/ptps.6.93.

[25] K. Greisen. 'Cosmic ray showers'. In: *Annual Review of Nuclear Science* 10.1 (1960), pp. 63–108. DOI: 10.1146/annurev.ns.10.120160.000431.

[26] T. Pierog et al. 'EPOS LHC: Test of collective hadronization with data measured at the CERN Large Hadron Collider'. In: *Phys. Rev. C* 92 (3 Sept. 2015), p. 034906. DOI: 10.1103/PhysRevC.92.034906.

[27] F. Riehn et al. 'A new version of the event generator Sibyll'. In: *PoS* ICRC2015 (2016), p. 558. DOI: 10.22323/1.236.0558. arXiv: 1510.00568 [hep-ph].

[28] S. Ostapchenko. 'Monte Carlo treatment of hadronic interactions in enhanced Pomeron scheme: QGSJET-II model'. In: *Phys. Rev. D* 83 (1 Jan. 2011), p. 014018. DOI: 10.1103/PhysRevD.83.014018.

[29] S. Schoo et al. 'A new analysis of the combined data from both KASCADE and KASCADE-Grande'. In: *PoS* ICRC2017 (2017), p. 339. DOI: 10.22323/1.301.0339.

[30] W. Apel et al. 'The spectrum of high-energy cosmic rays measured with KASCADE-Grande'. In: *Astroparticle Physics* 36.1 (2012), pp. 183–194. ISSN: 0927-6505. DOI: 10.1016/j.astropartphys.2012.05.023.

[31] D. Bird et al. 'The cosmic-ray energy spectrum observed by the Fly's Eye'. In: *The Astrophysical Journal* 424 (1994), pp. 491–502. DOI: 10.1086/173906.

[32] W. Hanlon. 'Telescope Array 10 Year Composition'. In: *PoS* ICRC2019 (2019), p. 280. DOI: 10.22323/1.358.0280.

[33] P. Abreu et al. 'Interpretation of the Depths of Maximum of Extensive Air Showers Measured by the Pierre Auger Observatory'. In: *JCAP* 02 (2013), p. 026. DOI: 10.1088/1475-7516/2013/02/026. arXiv: 1301.6637 [astro-ph.HE].

[34] The Pierre Auger Collaboration. 'The Pierre Auger Cosmic Ray Observatory'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 798 (2015), pp. 172–213. ISSN: 0168-9002. DOI: 10.1016/j.nima.2015.06.058.

[35] T. Huege on behalf of the Pierre Auger Collaboration. 'Radio detection of cosmic rays with the Auger Engineering Radio Array'. In: *EPJ Web Conf.* 210 (2019), p. 05011. DOI: 10.1051/epjconf/201921005011.

[36] B. Fuchs. 'The Auger Engineering Radio Array'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 692 (2012). 3rd Roma International Conference on Astroparticle Physics, pp. 93–97. ISSN: 0168-9002. DOI: 10.1016/j.nima.2012.01.058.

[37] S. de Jong. 'The Radio detection of inclined showers at the Pierre Auger Observatory'. In: *40th International Conference on High Energy Physics*. Dec. 2020. DOI: 10.22323/1.390.0829. arXiv: 2012.05044 [astro-ph.IM].

[38] A. Aab et al. 'Calibration of the underground muon detector of the Pierre Auger Observatory'. In: *Journal of Instrumentation* 16.04 (Apr. 2021). ISSN: 1748-0221. DOI: 10.1088/1748-0221/16/04/p04003.

[39] The Pierre Auger Collaboration. 'The Pierre Auger Observatory Upgrade - Preliminary Design Report'. In: (Apr. 2016). DOI: 10.2172/1250882. arXiv: 1604.03637 [astro-ph.IM].

[40] C. Meurer and N. Scharf on behalf of the Pierre Auger Collaboration. 'HEAT – a low energy enhancement of the Pierre Auger Observatory'. In: *Astrophysics and Space Sciences Transactions* 7.2 (May 2011), pp. 183–186. ISSN: 1810-6536. DOI: 10.5194/astra-7-183-2011.

[41] The Pierre Auger Collaboration. 'The Fluorescence Detector of the Pierre Auger Observatory'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 620.2 (2010), pp. 227–251. ISSN: 0168-9002. DOI: 10.1016/j.nima.2010.04.023.

[42] The Pierre Auger Collaboration. 'Calibration of the surface array of the Pierre Auger Observatory'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 568.2 (2006), pp. 839–846. ISSN: 0168-9002. DOI: 10.1016/j.nima.2006.07.066.

[43] S. Dasso and H. Asorey. 'The scaler mode in the Pierre Auger Observatory to study heliospheric modulation of cosmic rays'. In: *Adv. Space Res.* 49 (2012), pp. 1563–1569. DOI: 10.1016/j.asr.2011.12.028.

[44] The Pierre Auger Collaboration. 'Trigger and aperture of the surface detector array of the Pierre Auger Observatory'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 613 (Jan. 2010), pp. 29–39. DOI: 10.1016/j.nima.2009.11.018.

[45] A. Aab et al. 'Reconstruction of Events Recorded with the Surface Detector of the Pierre Auger Observatory'. In: *Journal of Instrumentation* 15.10 (Oct. 2020), P10021–P10021. DOI: 10.1088/1748-0221/15/10/p10021.

[46] J. Hersil et al. 'Observations of Extensive Air Showers near the Maximum of Their Longitudinal Development'. In: *Phys. Rev. Lett.* 6 (1 Jan. 1961), pp. 22–23. DOI: 10.1103/PhysRevLett.6.22.

[47] H. P. Dembinski et al. 'A likelihood method to cross-calibrate air-shower detectors'. In: *Astroparticle Physics* 73 (2016), pp. 44–51. ISSN: 0927-6505. DOI: 10.1016/j.astropartphys.2015.08.001.

[48] R. Smida on behalf of the Pierre Auger Collaboration. 'Scintillator detectors of AugerPrime'. In: *Proceedings of 35th International Cosmic Ray Conference — PoS(ICRC2017)*. Vol. 301. 2017, p. 390. DOI: 10.22323/1.301.0390.

[49] T. Suomijarvi on behalf of the Pierre Auger Collaboration. 'New electronics for the surface detectors of the Pierre Auger Observatory'. In: 301 (2017), p. 450. DOI: 10.22323/1.301.0450.

[50] A. Castellina on behalf of the Pierre Auger Collaboration. 'The dynamic range of the AugerPrime SurfaceDetector: technical solution and physics reach'. In: 301 (2017), p. 397. DOI: 10.22323/1.301.0397.

[51] L. Nellen. 'The observation of a muon deficit in simulations from data of the Pierre Auger Observatory'. In: *J. Phys. Conf. Ser.* 409.1 (2013). Ed. by A. D. Erlykin et al., p. 012107. DOI: 10.1088/1742-6596/409/1/012107.

[52] Z. Zong on behalf of the Pierre Auger Collaboration. 'First results from the AugerPrime Engineering Array'. In: 301 (2017), p. 449. DOI: 10.22323/1.301.0449.

[53] A. Taboada. 'Analysis of Data from Surface Detector Stations of the AugerPrime Upgrade'. In: *Proceedings of 36th International Cosmic Ray Conference — PoS(ICRC2019)*. Vol. 358. 2019, p. 434. DOI: 10.22323/1.358.0434.

[54] The Pierre Auger Collaboration. *CDASPublic Webpage*. URL: https://www.auger.org.ar/CDAS-Public/.

[55] S. Argirò et al. 'The offline software framework of the Pierre Auger Observatory'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 580.3 (2007), pp. 1485–1496. ISSN: 0168-9002. DOI: 10.1016/j.nima.2007.07.010.

[56] I. C. Maris et al. *Data Summary Trees and Shower Visualization for Reconstructed Auger Events*. Pierre Auger Collaboration Internal Note. 2006.

[57] URL: https://github.com/HDembinski/pyik.

[58] S. Messina and A. M. van den Berg. *Reconstruction of 'AERA'–triggered air showers*. Pierre Auger Collaboration Internal Note. 2013.

[59] S. Messina, J. van den Eijnden and A. M. van den Berg. *Energy Calibration for the 433 m SD Infill Array*. Pierre Auger Collaboration Internal Note. 2006.

[60] F. Meyer and F. Vernotte. *Time tagging tests in Besancon Observatory*. Pierre Auger Collaboration Internal Note. 2001.

[61] A. Ivakhnenko, V. Lapa and R. McDonough. *Cybernetics and Forecasting Techniques*. Modern analytic and computational methods in science and mathematics. American Elsevier Publishing Company, 1967. ISBN: 9780444000200.

[62] X. Glorot and Y. Bengio. 'Understanding the difficulty of training deep feedforward neural networks'. In: *Journal of Machine Learning Research - Proceedings Track* 9 (Jan. 2010), pp. 249–256.

[63] K. He et al. 'Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification'. In: *IEEE International Conference on Computer Vision (ICCV 2015)* 1502 (Feb. 2015). DOI: 10.1109/ICCV.2015.123.

[64] D. P. Kingma and J. Ba. 'Adam: A method for stochastic optimization'. In: *arXiv preprint arXiv:1412.6980* (2014).

[65] I. Goodfellow, Y. Bengio and A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[66] N. Srivastava et al. 'Dropout: A Simple Way to Prevent Neural Networks from Overfitting'. In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958.

[67] T. S. Cohen and M. Welling. 'Group Equivariant Convolutional Networks'. In: *CoRR* abs/1602.07576 (2016). arXiv: 1602.07576.

[68] D. Schattschneider. 'The plane symmetry groups: their recognition and notation'. In: *The American Mathematical Monthly* 85.6 (1978), pp. 439–450. DOI: 10.2307/2320063.

[69] E. Hoogeboom et al. *HexaConv*. 2018. arXiv: 1803.02108 [cs.LG].

[70] J. Luo et al. 'Hexagonal Convolutional Neural Networks for Hexagonal Grids'. In: *IEEE Access* 7 (2019), pp. 142738–142749. DOI: 10.1109/access.2019.2944766.

[71] D. Heck et al. *CORSIKA: a Monte Carlo code to simulate extensive air showers.* 1998.

[72] S. Agostinelli et al. 'Geant4—a simulation toolkit'. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (2003), pp. 250–303. ISSN: 0168-9002. DOI: 10.1016/S0168-9002(03)01368-8.

[73] C. R. Harris et al. 'Array programming with NumPy'. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2.

[74] F. Chollet et al. *Keras*. https://keras.io. 2015.

[75] M. Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. DOI: 10.5281/zenodo.4724125. URL: https://www.tensorflow.org/.

[76] G. Huang et al. *Densely Connected Convolutional Networks*. 2018. arXiv: 1608.06993 [cs.CV].

[77] I. Lhenry-Yvon and F. Salamida. *SD/FD Performance Status*. Pierre Auger Collaboration internal report. 2020.

[78] *Hybrid Xmax acceptance and resolution*. Pierre Auger Collabortation internal Wiki page. URL: https://www.auger.unam.mx/AugerWiki/Hybrid_Xmax_acceptance_and_resolution.

[79] E. J. Ahn et al. *Measurement of the Depth of Shower Maximum of Cosmic Rays above* 10^{18} *eV*. Pierre Auger Collaboration Internal Note. 2009.

[80] A. Aab et al. 'Deep-learning based reconstruction of the shower maximum X_{max} using the water-Cherenkov detectors of the Pierre Auger Observatory'. In: *JINST* 16.07 (2021), P07019. DOI: 10.1088/1748-0221/16/07/P07019.

Acknowledgments

> *I believe that when we leave a place, part of it goes with us and part of us remains. Go anywhere in these halls, when it is quiet and just listen. After a while you will hear the echoes of all of our conversations, every thought and word we've exchanged. Long after we are gone, our voices will linger in these walls for as long as this place remains. But I will admit that the part of me that is going will very much miss the part of you that is staying.*
>
> — Citizen G'Kar, *Babylon 5*

This thesis and the research presented would not have been possible without a number of people giving me their time and support.

First of all, I would like to thank Prof. Dr. Karl-Heinz Kampert for giving me the opportunity to join his working group and work on this interesting research topic. I am grateful for all his advice and him taking time out of even the busiest days when I had questions. I would also like to thank Prof. Dr. Antonio Bueno Villar for agreeing to be the second reviewer of this work. I want to thank Dr. Julian Rautenberg for aiding me in this research, always having an open ear and valuable feedback over the last years, and of course for all those boxes of cookies.

I want to give a big thank you to the Astros, for being so welcoming when I first joined the group from the 'terrible' city of Dortmund. Thank you for the great summer events, Christmas parties, coffee breaks, and weekend beers. I will always think of you when it hits 4 o'clock on a Friday.

To my friends from Dortmund - Max Kämper, Koray Önder, Jan Soedingrekso, Felix Föst, Daniela Röttges - without you I would not have made it to where I am now. From studying together to losing every single game of Mario Kart, I always enjoyed my time with you and hope we can meet for the annual Christmas market picture. Thank you to Martin Will, for always reminding me to "halt die Ohren steif c-.-ɔ", educating me about Haarmilch and Wirsing, and leaving Oktoberfest early to watch First Contact. And thank you to Sandra Koch, who has seen me through every high and low, you only considered me half crazy when I decided to study physics and you have helped me keep an almost healthy physics/life balance.

None of this would have been possible without my parents and brother. I want to thank them for always supporting me and most importantly, for instilling in me a love for science fiction. From letting me watch Star Trek and Babylon 5, to playing starfleet officers with my brother. If it was not for them and these TV shows that shaped my childhood, I would not have developed an interest in science, let alone studied physics.

The biggest thank you goes to my husband Eric, for always encouraging me to keep going and being willing to give advice and bounce ideas back and forth at any time of day, especially when it came to designing the network architecture. Without your support I would not have been able to climb this mountain. Thank you for eating all those clowns for breakfast to cheer me up when things got stressful, for always being there, and for the endless supply of coffee. I cannot wait to see what the future holds for us.

Declaration of Authorship

English

I hereby declare that the thesis submitted is my own unaided work. All direct or indirect sources used are acknowledged as references.

This thesis was not previously presented to another examination board and has not been published.

Deutsch

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Wuppertal, den _____ _____

(Sonja Mayotte)